AD-A274 132

0

AFIT/GA/ENG/93D-01

MULTIPLE MODEL ADAPTIVE CONTROL OF A LARGE FLEXIBLE SPACE STRUCTURE WITH PURPOSEFUL DITHER FOR ENHANCED IDENTIFIABILITY

THESIS

James Alan Fitch
Captain, USAF

AFIT/GA/ENG/93D-01

DTIC
S ELECTE D
DEC 23 1993
E

Approved for public release; distribution unlimited

93 12 22 101

93-30988

14286

AFIT/GA/ENG/93D-01

MULTIPLE MODEL ADAPTIVE CONTROL OF A LARGE FLEXIBLE SPACE STRUCTURE WITH PURPOSEFUL DITHER FOR ENHANCED IDENTIFIABILITY

THESIS

Presented to the Faculty of the Graduate School of Engineering
of the Air Force Institute of Technology
Air University
In Partial Fulfillment of the
Requirements for the Degree of
Master of Science in Astronautical Engineering

DTIC
COPY
INSPECTED
6

Accesion For
NTIS CRA&I ☒
DTIC TAB ☐
Unannounced ☐
Justification ....
By ....
Distribution /
Availability Codes
Dist | Avail and / or Special
A-1

James Alan Fitch, B.S.A.E
Captain, USAF

December 1993

Approved for public release; distribution unlimited

Preface

The main purpose of this thesis research is to apply dithering techniques for enhanced identifiability of uncertain parameters using moving-bank Multiple Model Adaptive Estimation and Control (MMAE/MMAC) algorithms. These techniques are applied to the SPICE 2 flexible space structure in order to quell induced structural vibrations. Results of this research indicate that purposefully constructed dither signals can enhance parameter identification to a significant degree.

I must extend great thanks and gratitude to Dr. Peter Maybeck for his patience and thoughtful guidance. Dr. Maybeck provided the utmost encouragement and enthusiasm throughout this endeavor, without which I would not have been able to succeed. I also wish to thank Capt Greg Schiller for all his help with System Build and Word. I would not have been able to make it without you Greg. A bit of thanks must go to Capt Robert Nielsen for always having his Estimation and Control texts available for my perusal in the Nav Lab. The biggest thank you, however, must go to my wife, Ginger, and son, Orren, for giving me the time and space needed to complete this work. Thank you Ginger for keeping your sanity in tack and for hanging in there until the end.

James A. Fitch

Table of Contents

page

Preface ii

Table of Contents iii

List of Figures viii

List of Tables xii

Abstract xiii

1. Introduction 1-1

  1.1 Notation 1-6

  1.2 Background 1-6

    1.2.1 System Model 1-6

    1.2.2 Multiple Model Adaptive Estimation 1-8

    1.2.3 Moving-Bank MMAE 1-9

    1.2.4 Moving-Bank MMAC 1-11

  1.3 Past Research 1-13

    1.3.1 MMAE/MMAC Techniques 1-13

    1.3.2 Dithering Applications 1-16

  1.4 Objectives 1-17

  1.5 Scope 1-18

  1.6 Approach 1-18

  1.7 Summary 1-20

2. Development of Fundamental Concepts 2-1

  2.1 Introduction 2-1

page

2.2 Kalman Filter (Bayesian Estimation) 2-1

2.3 MMAE 2-7

 2.3.1 Bayesian MMAE 2-7

 2.3.2 Enhancements to Bayesian Estimation 2-11

2.4 Moving-Bank MMAE Development 2-13

 2.4.1 Moving the Bank 2-14

  2.4.1.1 Residual Monitoring 2-14

  2.4.1.2 Parameter Position Estimate Monitoring 2-15

  2.4.1.3 Probability Monitoring 2-15

 2.4.2 Expanding the Bank 2-16

 2.4.3 Contracting the Bank 2-16

 2.4.4 Initialization of New Elemental filters 2-17

2.5 Stochastic Controller Development 2-18

2.6 MMAC 2-22

 2.6.1 MMAC Control 2-23

 2.6.2 "Modified" MMAC Control 2-23

 2.6.3 MMAC with MAP Control 2-24

 2.6.4 Single Fixed-Gain Control 2-24

 2.6.5 Single Changeable-Gain Control 2-25

 2.6.6 "Modified" Single Changeable-Gain Control 2-26

2.7 Dithering Techniques 2-27

page

2.7.1 Sinusoidal Dither 2-28

2.7.2 Dither Frequency 2-28

2.7.3 Wide Band Noise Dither 2-30

2.8 Mathematical Modeling Methods 2-32

2.7.1 Physical Coordinate Form 2-33

2.7.2 Modal Coordinate Form 2-35

2.7.3 Modal Reduction Technique 2-38

2.9 Summary 2-43

3. System Development 3-1

3.1 Introduction 3-1

3.2 SPICE Structure 3-1

3.2.1 Physical Description 3-2

3.2.2 Actuators and Sensors 3-2

3.3 System Mathematical Model Description 3-4

3.3.1 Disturbances 3-5

3.3.2 Proof Mass Actuators (PMAs) 3-7

3.3.3 Structure 3-10

3.3.3.1 Flexible Dynamics 3-10

3.3.3.2 Rigid-body Dynamics 3-13

3.3.4 Accelerometers 3-15

3.4 Truth Model and Filter Model 3-17

page

3.5 Summary 3-18

4. Simulation 4-1

4.1 Introduction 4-1

4.2 Monte Carlo Analysis 4-1

4.2.1 Error Vector Formulations 4-3

4.2.1 Error Vector Statistics 4-4

4.3 Simulation Software 4-5

4.3.1 Preprocessor 4-5

4.3.2 Processor 4-6

4.3.3 Post Processor 4-8

4.4 Analysis Plan 4-8

4.4.1 Dither Frequency Selection 4-8

4.4.2 Improved Accelerometer Measurements 4-9

4.4.3 Parameter Space Discretization (Sensitivity Analysis) 4-11

4.5 Summary 4-12

5. Results 5-1

5.1 Introduction 5-1

5.2 MMAE/MMAC "Best Case" Performance 5-2

5.2.1 Single Filter Baseline 5-2

5.2.2 Multiple Model Baseline 5-2

5.3 Parameter Estimation Performance 5-3

page

5.3.1 Baseline (No Dither) 5-4

5.3.2 Dither Signal Enhancement Performance 5-4

5.3.3 Rigid-body Motion Enhancement Performance 5-6

5.4 New Parameter Space Discretization 5-7

5.4.1 Baseline (No Dither) 5-8

5.4.2 Dither Signal Enhancement Performance 5-8

5.5 Summary of ME/A vs ME/I Performance 5-9

5.6 Increased Measurement Precision Performance 5-9

5.7 Reduced-Order Design Performance 5-10

5.8 Summary 5-11

6. Conclusions and Recommendations 6-1

6.1 Introduction 6-1

6.2 Conclusions 6-1

6.3 Recommendations 6-3

Appendix A: MMAE, MMAC, and Parameter Estimation Plots A-1

Bibliography BIB-1

VITA VITA-1

List of Figures

page

Figure 1-1. Multiple Model Adaptive Estimator[20:132] 1-3

Figure 1-2. Multiple Model Adaptive Controller [21: 254] 1-5

Figure 1-3: SPICE Space Structure 1-7

Figure 1-4. Full-Bank MMAE 1-9

Figure 1-5. Moving Bank MMAE Fine Discretization 1-11

Figure 1-6. Moving-Bank MMAE Coarse Discretization 1-12

Figure 2-1. Single Fixed-Gain Controller 2-25

Figure 2-2. Single Changeable-Gain Controller 2-26

Figure 2-3. Modified Single Changeable-Gain Controller 2-27

Figure 2-4. Block Diagram for Wide Band Noise 2-31

Figure 2-5. PSD Plot for Wide Band Noise 2-31

Figure 3-1. SPICE Structure [14: III-14] 3-3

Figure 3-2. Flexible SPICE Structure 3-4

Figure 3-3. SPICE System Block Diagram [2: 3-5] 3-5

Figure 3-4. Disturbance Model Block Diagram [2: 3-6] 3-6

Figure 3-5. PMA Block Diagram 3-8

Figure 3-6. Rigid-Body Rotational Motion 3-13

Figure 3-7. Accelerometer Design 3-15

Figure 4-1. (a) Estimator Simulation, and (b) Controller Simulation [29] 4-2

Figure A-1. Single Filter Estimation and Control Baseline (Open Loop) A-2

page

Figure A-2. Single Filter Estimation and Control Baseline (Closed Loop) A-3

Figure A-3. Multiple Filter Estimation and Control Baseline (Closed Loop, ME/A) A-4

Figure A-4. Multiple Filter Estimation and Control Baseline (Closed Loop, ME/I) A-5

Figure A-5. 7-Point Parameter Space: True Parameter Jump, No Dither (ME/A) A-6

Figure A-6. 7-Point Parameter Space: True Parameter Jump, No Dither (ME/I) A-6

Figure A-7. 7-Point Parameter Space: Parameter Variation, No Dither (ME/A) A-6

Figure A-8. 7-Point Parameter Space: Parameter Variation, No Dither (ME/I)............ A-7

Figure A-9. 7-Point Parameter Space: Swept Sine Wave Dither, 29-33 Hz (ME/A) A-7

Figure A-10. 7-Point Parameter Space: Sine Wave Dither (ME/A) A-7

Figure A-11. 7-Point Parameter Space: Sine Wave Dither (ME/I).............................. A-8

Figure A-12. 7-Point Parameter Space: Square Wave Dither (ME/A) A-8

Figure A-13. 7-Point Parameter Space: Square Wave Dither (ME/I).......................... A-8

Figure A-14. 7-Point Parameter Space: Wide Band Noise Dither, 7-33 Hz (ME/I)...... A-9

Figure A-15. 7-Point Parameter Space: Wide Band Noise Dither, 30-33 Hz (ME/A) A-9

Figure A-16. 7-Point Parameter Space: Wide Band Noise Dither, 30-33 Hz (ME/I).................. A-9

Figure A-17. 7-Point Parameter Space: Rigid-body Dither (ME/A).......................... A-10

Figure A-18. 7-Point Parameter Space: Rigid-body Dither (ME/I) A-10

Figure A-19. 29-Point Parameter Space: No Dither (ME/A) A-10

page

Figure A-20. 29-Point Parameter Space: No Dither (ME/I) A-11

Figure A-21. 29-Point Parameter Space: Sine Wave Dither (ME/A) A-11

Figure A-22. 29-Point Parameter Space: Sine Wave Dither (ME/I) A-11

Figure A-23. 29-Point Parameter Space: Square Wave Dither (ME/A) A-12

Figure A-24. 29-Point Parameter Space: Square Wave Dither (ME/I) A-12

Figure A-25. 29-Point Parameter Space: Wide Band Noise Dither, 30-33 Hz (ME/A) A-12

Figure A-26. 29-Point Parameter Space: Wide Band Noise Dither, 30-33 Hz (ME/I) A-13

Figure A-27. 29-Point Parameter Space: Rigid-body Dither (ME/A) A-13

Figure A-28. 29-Point Parameter Space: Rigid-body Dither (ME/I) A-13

Figure A-29. Single Filter Estimation and Control, 26-Mode Reduced-Order (Closed Loop) A-14

List of Tables

page

Table 3-1. Modal eigenvalues and natural frequencies for the first 30 modes 3-30

Table 3-2. Top 30 Component Cost values and associated modes 3-32

Table 5-1. Filter/Controller Model Performance Results.. 5-9

Table 5-2. MMAC Control Performance Results.. 5-24

Abstract

Dithering techniques for enhancing uncertain parameter identification with moving-bank Multiple Model Adaptive Estimation (MMAE) and Control (MMAC) algorithms are analyzed in this thesis. The dithering techniques and multiple-model adaptive algorithms are applied to the SPICE 2 flexible space structure.

The dithering techniques studied include purposefully constructed square wave, sine wave (fixed and swept frequency) and wide-band noise wave forms. Purposeful rigid-body slew commands are also used in order to excite the structure's flexible bending modes. Dither inputs into the structure are performed in an effort to enhance the open-loop identifiability of the uncertain parameter, namely a scalar multiplier on the undamped natural frequencies of the flexible bending modes. Correct identification of this parameter enables the MMAE and MMAC algorithms to quell any vibrations induced into the structure, in the face of varying parameter values.

The results of this study indicate that purposefully constructed dither signals do enhance parameter identification significantly. The parameter identification enhancement due to rigid-body slew command input was shown to be less effective than that with sinusoid and wide band noise dither inputs, however.

MULTIPLE MODEL ADAPTIVE CONTROL OF A LARGE FLEXIBLE SPACE STRUCTURE WITH PURPOSEFUL DITHER FOR ENHANCED IDENTIFIABILITY

1. Introduction

Kalman filters are very useful in providing state estimates for systems driven by known inputs and white Gaussian noise. In any state estimation problem, however, there is the underlying problem of obtaining a correct dynamics model for the filter. It may well be possible to obtain a model for a typical system or one for an assumed set of conditions, but real systems do not stay constant. The dynamic parameters describing a system will undoubtedly vary over time. (We term these *uncertain parameters*). Thus, the problem is a single Kalman filter model may not be robust enough to meet desired specifications.

There are many methods to overcome the uncertain parameter problem, and these are thoroughly discussed in [20: Chapter 10]. One particular method will be addressed here: Multiple Model Adaptive Estimation (MMAE) [20: 129-136]. With this method are developed various (multiple) Kalman filter models, each based on a different value of a set of uncertain parameters. For example, the uncertain parameters for a second order system might be the damping ratio and the undamped natural frequency. If these were each known to vary over a particular range, then multiple filters could be constructed, each with a different value for the damping ratio and undamped natural frequency.

With the multiple models developed, each is (conceptually) used as the basis for a single elemental filter to form a "bank" or column of filters. The output of each filter (the state estimate based on a particular parameter value) is given a probability that it is the estimate based on the correct parameter value. This probability is calculated from a history of measurement information and from known system characteristics and noise statistics (to which adaptation is not required). Each probabilistically weighted estimate is then summed, resulting in a probabilistically weighted average of the single filter state estimates.

The structure of the Multiple Model Adaptive Estimator can be seen in Figure 1-1. As shown, the bank consists of K separate Kalman filters, each based on particular parameter value from the set $\{\mathbf{a}_1, \ \mathbf{a}_2, \ . \ . \ . \ . \ , \ \mathbf{a}_K\}$. Each filter obtains the current measurement input, $\mathbf{z}$, and from this outputs a residual, $\mathbf{r}_k$, and a state estimate, $\hat{\mathbf{x}}_k$. The residual (the difference between the measurement and best prediction of that measurement based on previous data) is used to compute the conditional probability, which is then used to form the probabilistically weighted average of the K state estimates.

The MMAE presented thus far is considered a "fixed-bank" MMAE, and as it appears in Figure 1-1, the MMAE is based upon a finite number of parameter points, i.e., a finite number of Kalman filters. This finite size of the parameter set used for the MMAE is necessary because, if the parameter set were to consist of all possible points from a continuous space, the number of filters required would be infinite. Thus, only a finite number of points in the appropriate range of values are chosen for the parameter set. This is then called a discretized parameter space. Now, even with a finite number of Kalman filters within this discretized parameter space, the number of filters can be quite large and thus produce a considerable computational burden.

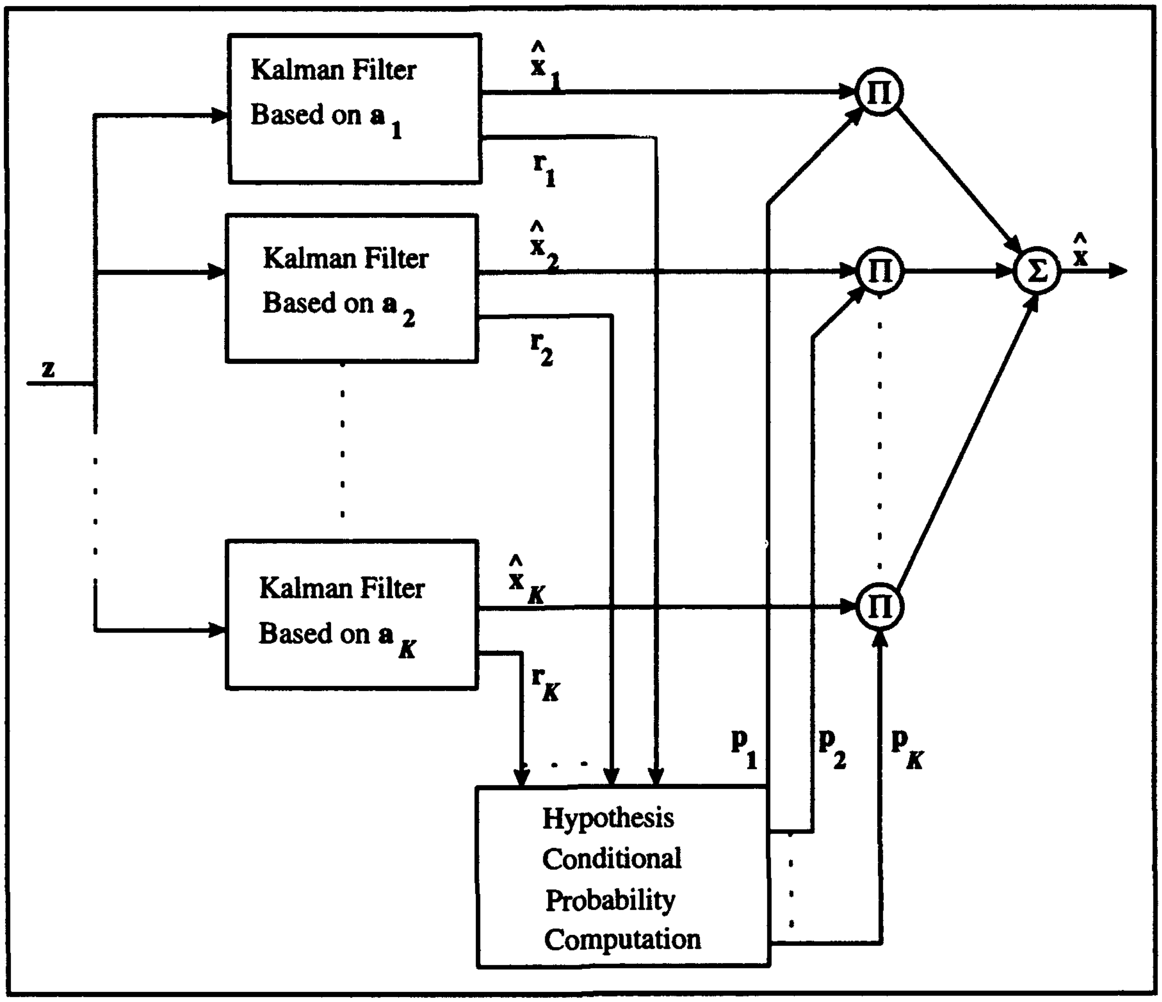

Figure 1-1. Multiple Model Adaptive Estimator[20:132]

To lessen this burden, the concept of a subset of the original parameter set is introduced. This subset is a small grouping within the parameter space, always centered on the current estimate of the true parameter value. This small subset can then “move” within the entire parameter space, depending on the location of the parameter estimate, and thus the term “moving bank” [17]. A discussion on the moving bank will be covered in more detail in Section 1.2.3.

With the state estimate in hand, we will now discuss the controller to be used in this research. Corresponding to linear Kalman filters driven by white Gaussian noise, a controller based on a Linear model with a Quadratic cost criterion driven by white Gaussian noise (LQG) will be used. This stochastic controller will utilize the same K state estimates produced by the MMAE. Each separate state estimate is multiplied with a corresponding controller gain of a deterministic optimal regulator, $-\mathbf{G}_c^*$, to synthesize the optimal LQG controller for that specific parameter vector value. Just as there are K separate Kalman filters, each based on the parameter value, $\mathbf{a}_k$, there are K separate controller gains, each based on the parameter value, $\mathbf{a}_k$. The structure of this Multiple Model Adaptive Controller (MMAC) can be seen in Figure 1-2. As shown, the controller gains are cascaded to form control inputs, $\mathbf{u}_k$. The same probabilities used in the MMAE are used here to form a probabilistically weighted average control input, $\mathbf{u}$.

The research involved in this thesis is a direct continuation of the work of Gustafson [2, 3]. Gustafson applied the moving-bank MMAE and MMAC concepts to a large flexible structure developed at Phillips Laboratory, Kirtland AFB, NM (see Figure 1-3). This structure is part of a **SP**ace **I**ntegrated **C**ontrols **E**xperiment (SPICE). Section 1.2.1 gives a brief description of the SPICE model. At the time of Gustafson's work, the second model had been developed (SPICE 2). The primary goal of this thesis is to develop an MMAE/MMAC to quell any vibrations which are induced into the structure, despite uncertain parameters. Earlier contractor work [14] yielded a controller design that would almost meet design specifications if the parameters were assumed to be correctly known, but which caused closed-loop system instabilities if the real parameter values varied by as little as 2% from the assumed values. Gustafson did accomplish this goal, but the desired quelling specifications set by Phillips Laboratory were not met. (Phillips Laboratory requires the Line-of-Sight, or LOS, deviation to be within 1 μ-radian.) The work of this thesis will be to apply enhancements to the existing models put forth by

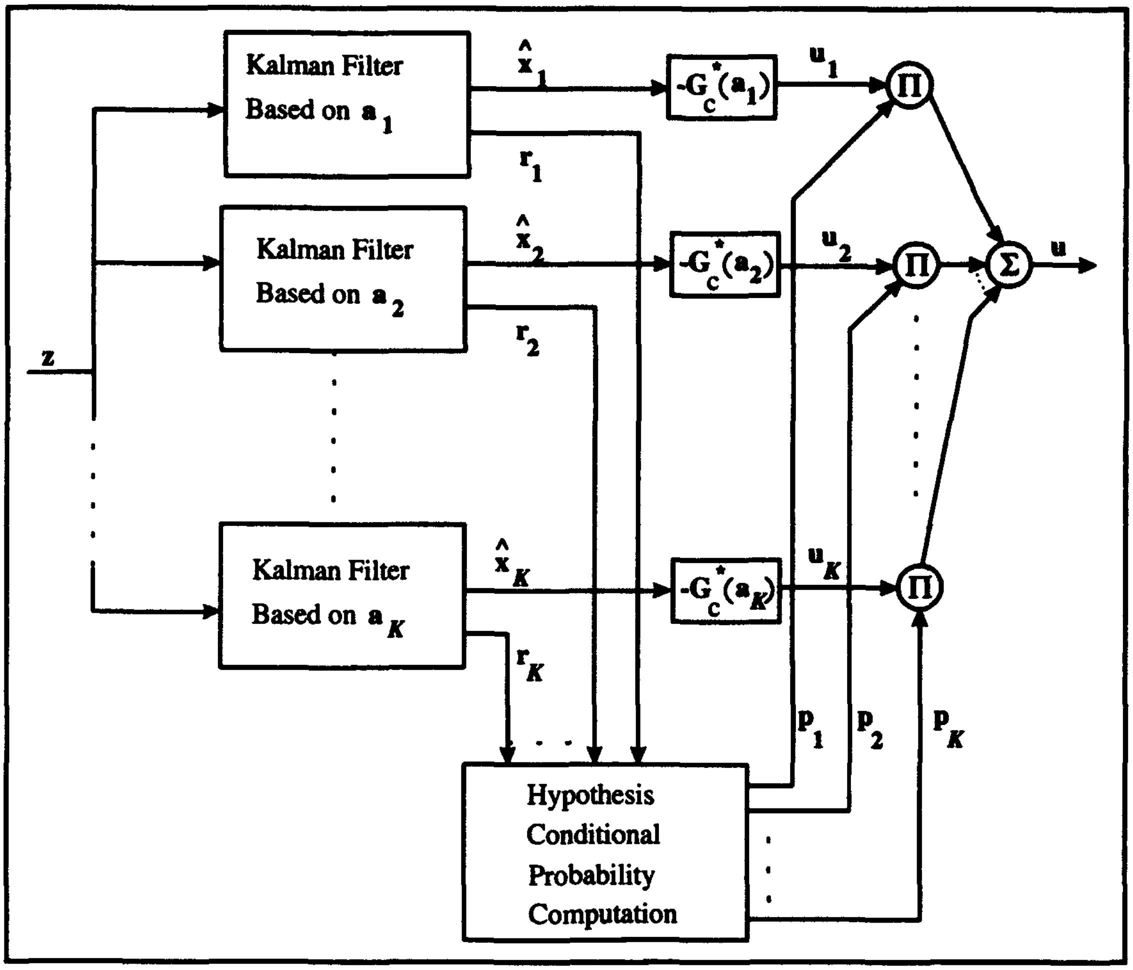

Figure 1-2. Multiple Model Adaptive Controller [21: 254]

Gustafson in order to determine what, if any, improvement in the MMAE/MMAC can be made towards meeting the desired quelling specifications. These enhancements include:

- Use dithering techniques to enhance uncertain parameter identifiability, prior to engaging closed-loop control
- Investigate the effects of purposeful rigid-body slew commands in regard to enhancing uncertain parameter identifiability
- Increase the measurement sensor precision
- Investigate an alternative parameter space discretization

1.1 Notation

The notation used in this thesis is consistent with that of [19]. *Scalars* in either upper or lower case letters are denotes by italic type. *Deterministic vectors and matrices* are denoted by boldface roman lower and upper case letters, respectively, e.g., $\mathbf{x}$ and $\mathbf{A}$. *Stochastic vectors and matrices* are denoted by boldface sans serif lower and upper case letters, respectively, e.g., $\boldsymbol{\mathsf{x}}$ and $\boldsymbol{\mathsf{A}}$. Similar to deterministic vectors, *realizations* of random variables are denoted by boldface roman, e.g., $\mathbf{x}$.

1.2 Background

This section presents an introduction to four areas, the concepts for which will be fully developed in Chapters 2 and 3. These areas are: (1) the system model, (2) Multiple Model Adaptive Estimation (MMAE), (3) Moving-Bank MMAE and (4) Moving-Bank Multiple Model Adaptive Control (MMAC).

1.2.1 System Model

The full physical and mathematical description of the SPICE 2 model will be presented in Chapter 3. Physically, the SPICE 2 structure is 6.19 meters wide at the base and is 8.14 meters tall [14]. In the simplest of terms, the space structure can be described as a tripod structure with a large mirror assembly at its base and a small mirror assembly at the top. In a realistic environment, the structure will experience vibrations (the three tripod legs will vibrate, causing the two mirrors to be out of alignment; high frequency oscillations of the stiff mirror assemblies will also take place, but with much smaller amplitude and effect on line-of-sight). In order to quell these vibrations, actuators are used which are located on the tripod legs and around the base of the structure. In order

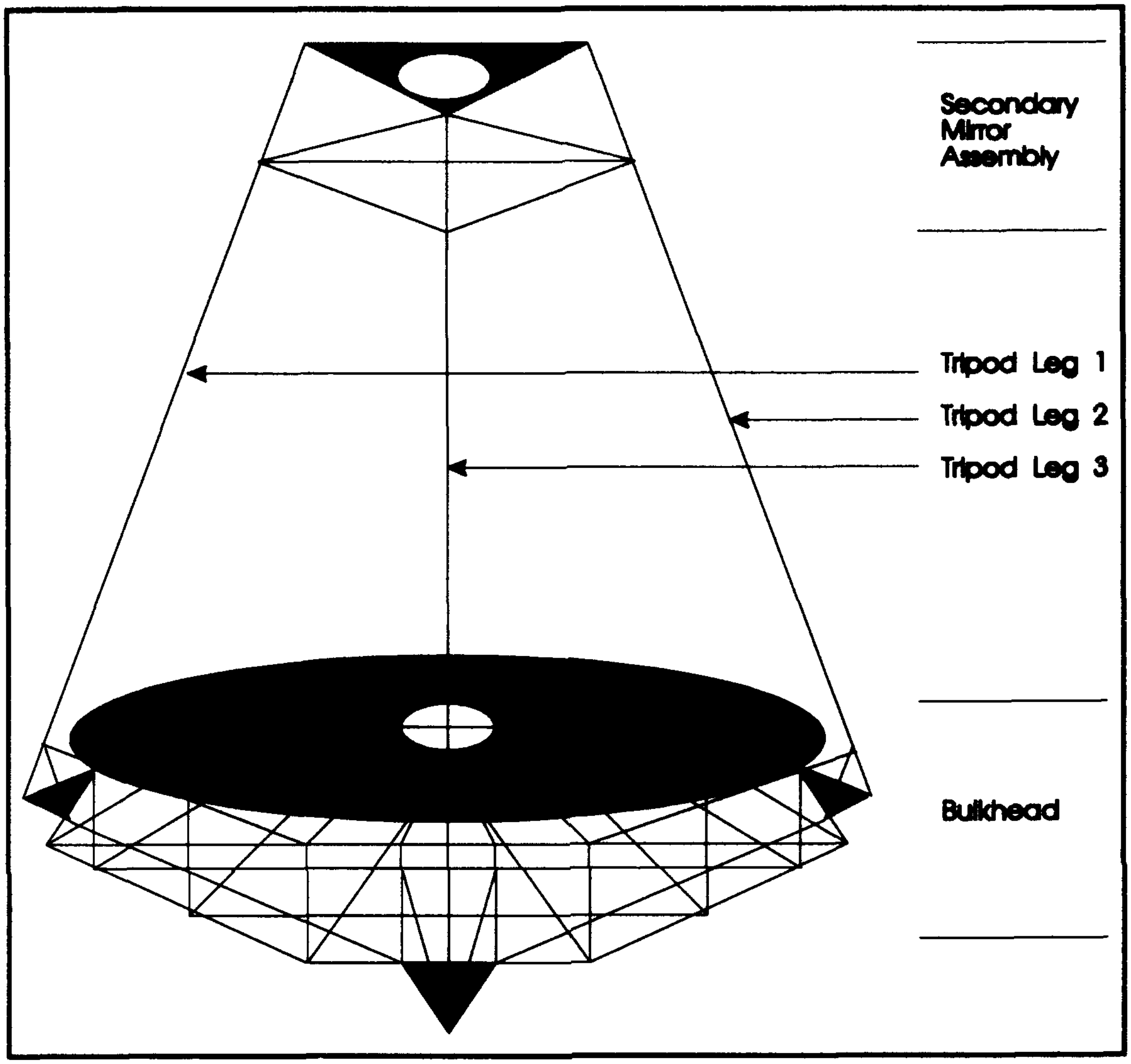

Figure 1-3: SPICE Space Structure

for this quelling of the structural vibrations to be accomplished, the true structural parameters must be accurately estimated (done by the MMAE). The parameters to be estimated consist of the undamped natural frequencies, ω_n, of the structural bending modes (see Chapter 2). Previous work using a simpler structural model used the undamped natural frequencies, ω_n, <u>and</u> the damping ratios, ζ, of the structural bending modes [1, 2, 3, 5, 7, 8, 11, 12, 22, 23, 26, 29, 32]. However, Gustafson [2: Chapter 5 pg.

36] discovered that, for the SPICE 2 model, adaptation to the damping ratio parameter was not needed and so is not used in this research.

1.2.2 Multiple Model Adaptive Estimation

As indicated in the introductory paragraphs, the need for an adaptive estimator comes from the problem of uncertain parameters in the system model from which a single Kalman filter might be formed. The multiple model approach utilized here involves a set of parameters representing various discrete values each parameter can take on in a real environment. For the discussion that follows, an example parameter space is formed using the undamped natural frequency and damping ratio as the two uncertain parameters. The discretized parameter space is formed by assuming a range of values for each of the two parameters. For example, let ω_n and ζ each take on 10 different values. Then 100 combinations of ω_n and ζ would be formed ($K = 100$). In this case there would also be 100 Kalman filters, each corresponding to a different combination of ω_n and ζ. This 100-point parameter space can be thought of existing in a 2-dimensional grid. This idea is illustrated in Figure 1-4.

This two-dimensional layout of the filters is important because is allows us to visualize the K Kalman filters as a bank which surrounds the location of the true parameter value (or actually a best estimate of that location). The MMAE algorithm is based on utilizing the residual information from each filter and the assumption that the filter with the smallest residual will most likely be the filter with parameter values that are closest to the true values [20: 133]. The residuals, $\mathbf{r}_k$, are used to produce a probability weighting factor. The smaller the residual, the larger the probability which is assigned to that filter's output. Then, as indicated in the introduction, this probability is used to form the probabilistically weighted state estimate. Figure 1-1 illustrates the pure Bayesian

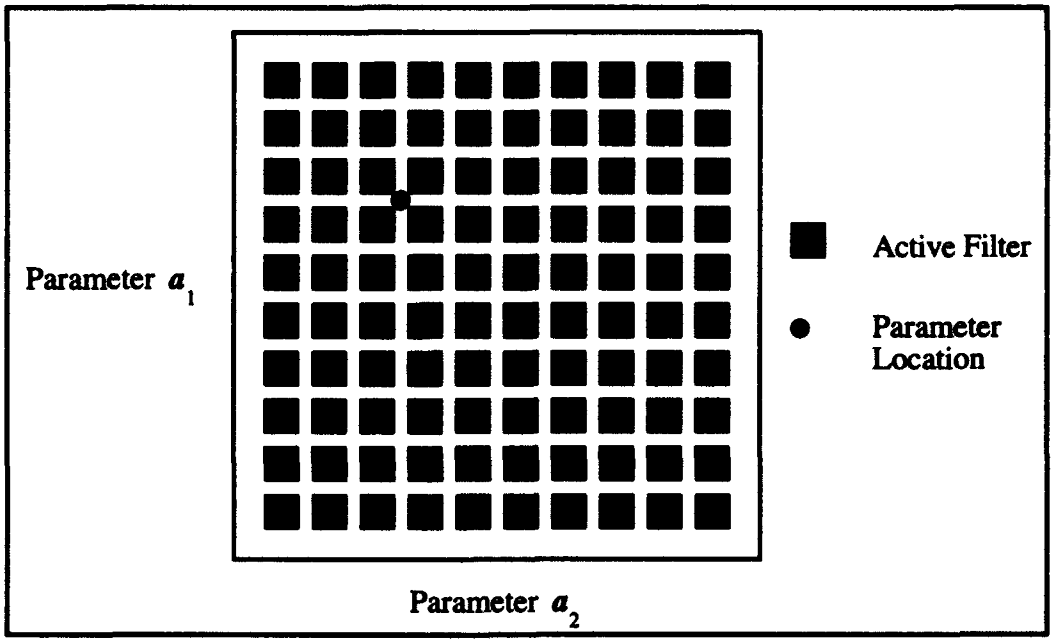

Figure 1-4. Full-Bank MMAE

formulation of the state estimate. There are, however, other methods of utilizing the residuals in calculating the probabilities and forming the probabilistically weighted state estimate. The four methods of performing this are: (1) Bayesian, (2) Maximum A Posteriori (MAP), (3) Bayesian with a Maximum Entropy with Identity Covariance (ME/I) computation, and (4) MAP with ME/I. These four methods will be fully discussed in Chapter 2.

1.2.3 Moving-Bank MMAE

The example parameter space considered in this discussion requires 100 Kalman filters to be run simultaneously. Obviously, this is would be quite a burden for any on-line computer application. As indicated in the introduction, the concept of a moving-bank of filters is introduced in order to alleviate this computational burden. The number of filters required in this smaller bank is that which will surround the parameter value. For our two-dimensional parameter space, nine filters have typically been used. This configuration allows a small but reasonable number of filters to perform the task of state and parameter estimation with the added benefit of having one *central* point for the basis of an elemental filter. In general we say the smaller moving-bank consists of J filters (parameter points), where $J < K$. The arrangement of parameter points discussed to this point is called *finely discretized* (all discrete points are adjacent to each other). Now, when the true parameter values vary, within the confines of the defined parameter space, the small subset of filters will "move" in order to keep the computed estimate of the true parameter value centered within the bank. This moving-bank concept is illustrated in Figure 1-5. As the parameter values change and "move" to a new location in the parameter space, so does the bank of filters. What is actually taking place is the activating and deactivating the appropriate Kalman filters, such that the bank is centered properly.

The moving-bank should be able to track a slowly varying parameter in this manner. However, should the parameter "jump" in value a great distance from its original location in the parameter space, a new logic must be implemented. If the true parameter value changes to a location in the parameter space outside the "bounds" of the finely discretized bank, all filters associated with the finely discretized bank will have large residuals, and the bank configuration must change to a *coarse discretization*, which uses filters along the outer edges of the parameter space. In this way the true parameter value will be encompassed by the bank. Once the new location of the parameter value is found, the bank can be finely discretized again. The coarse discretization concept is illustrated in Figure 1-6. Also note that the coarse discretization can be used during an initial acquisition when the parameter value is not known.

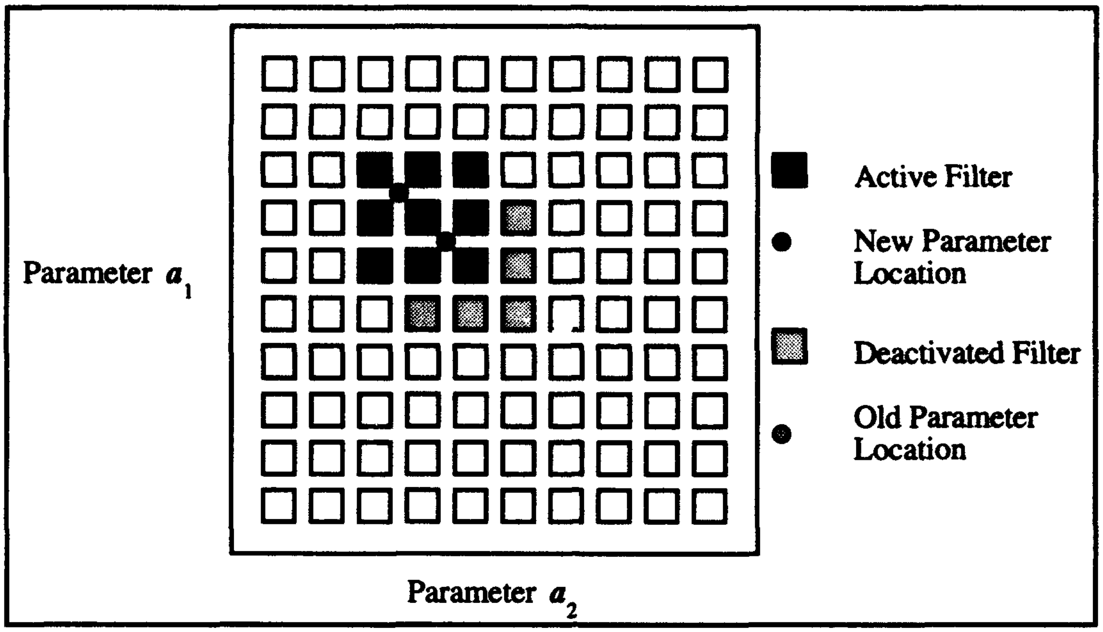

Figure 1-5. Moving Bank MMAE Fine Discretization

Just as there are several methods of implementing the MMAE, there are several methods for the decision-making logic of when and in what direction to move the bank, and when to expand and contract the bank. These methods are: (1) Residual Monitoring, (2) Parameter Position Estimate Monitoring, (3) Probability Monitoring, and (4) Parameter Estimation Error Covariance Monitoring. These methods will be fully discussed in Chapter 2.

1.2.4 Moving-Bank MMAC

The LQG stochastic controller will be implemented in the form of a regulator since the desired quelling action necessitates controlling all the structural states to zero [18, 21: 19-20]. Rigid-body states are not included in the discussion of the Kalman filter or the stochastic controller, since the purpose of the filter/controller is to quell the vibrating

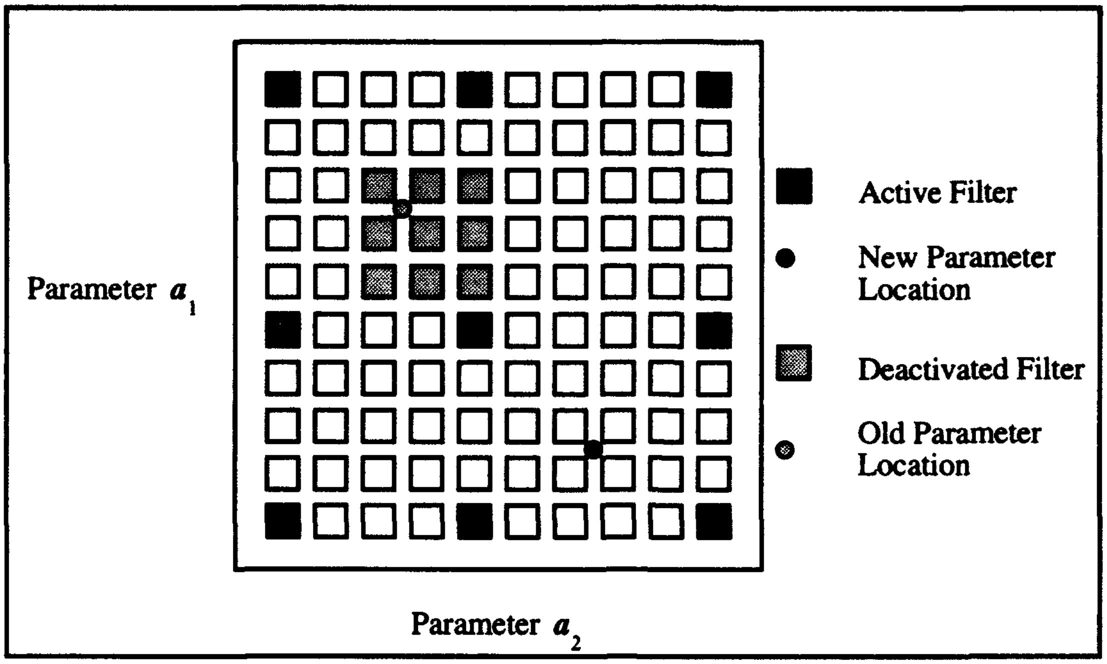

Figure 1-6. Moving-Bank MMAE Coarse Discretization

motion of the structure, not the rigid-body motion. A truth model rigid-body motion will be discussed in Chapters 2, 3 and 4.

The LQG stochastic controller used in this research is based on the *certainty equivalence property* [21: 17]. That is, the controller gain, $\mathbf{G}_c^*$, is formed assuming that the optimal gain calculated for perfectly known states is the same optimal gain to be used with state estimates. Each control gain, as shown in Figure 1-2, is calculated for each of the K parameter values, just as with the K Kalman filters in the MMAE design.

The combination of the MMAE output with the multiple-model LQG controller can take on many different forms. Figure 1-2 illustrates a pure MMAC structure with each Kalman filter cascaded with a single LQG controller that makes the same parameter value assumption as the corresponding filter. There are slight variations of this method called "Modified" MMAC Control and MMAC with MAP control. The pure MMAC and "Modified" MMAC methods are referred to as Bayesian because their output is probabilistically weighted using the hypothesis probability calculation, as shown in Figure 1-2. The MMAC with MAP method is a form of MMAC, but instead of using each conditional probability and elemental controller to form a probability-weighted average (as with the Bayesian form), this method picks the probability with the largest value and uses the associated single elemental controller to form the output control.

Other forms of multiple model control exist and are termed *MMAE based* control. These methods differ from the MMAC forms in the way the state estimates are utilized and/or in the way the final control output is formed. The *MMAE based* methods of control are: Single Fixed-Gain Control, Single Changeable-Gain Control, and "Modified" Single Changeable-Gain Control. Each of these methods of control will be discussed in Chapter 2.

1.3 Past Research

This section will present a brief summary of research accomplished in the areas of: (1) applying MMAE/MMAC methods to flexible structures and (2) applying dithering techniques to enhance parameter identifiability.

1.3.1 MMAE/MMAC Techniques

Research was begun by Hentz [5, 22] in the area of multiple model filters applied to the problem of quelling vibrations in a flexible structure. Hentz's research showed that the moving-bank, versus the fixed- or full-bank, adaptive filter and controller were viable solutions. The moving-bank MMAE/MMAC reduces the computational load substantially while maintaining adequate performance. Hentz showed this load reduction to be an order of magnitude lower [5]. Thus, a foundation was laid for continued research in this area.

Filios' [1] research indicated that ambiguity function analysis can provide insight as to which (if any) unknown parameters need to be estimated. This insight can be very useful when approaching a new system for the first time.

Karnick [7, 8] applied the MMAE/MMAC methods to a more complex model than had been done previously, namely that of a two-bay truss structure attached to a spacecraft. Karnick's algorithms had problems, though, correctly identifying the uncertain parameters, due to the sensors being too imprecise. However, his efforts did give accurate state estimation.

Lashlee [11, 12] followed up on the problems encountered by Karnick. He found that with a better-tuned filter and more accurate measurement data, the parameters and states could be accurately estimated. Basically, if there is too much measurement noise, the multiple model algorithm cannot distinguish between good and bad models of the uncertain parameters. Lashlee also found that the parameter space discretization plays an important role in the parameter identification performance.

Van Der Werken [32] applied the six-state filter from Lashlee's research to a higher-order "truth model" for simulation (24-state truth model versus a 6-state filter model). In his robustness study, he showed that the MMAE and MMAC performed poorly when comparing truth and filter model bending mode states directly.

Schore [23, 29] continued Van Der Werken's robustness study. His approach was to compare the truth and filter model representations of the flexed space structure at various physical locations on the structure. That is, he compared the position and velocity (the shape of the structure) at the two ends and at the midpoint of the two-bay truss, truth versus filter [29: Chapter 1 pg. 22]. Using this method, he showed that the reduced order filter model did not seriously degrade the performance of the estimator or controller. In fact, his MMAC maintained very adequate control in the face of an unmodeled disturbance.

Moyle's [26] research concentrated on streamlining the software that had been developed thus far and on further filter and controller tuning to enhance performance. Moyle's research showed marked performance improvement of the MMAE/MMAC, when compared against an artificially informed benchmark (an algorithm that was artificially provided knowledge of the true parameter). This performance was obtained by implementing a Maximum Entropy with Identity (ME/I) covariance computation approach (see Section 2.3.2) to parameter estimation along with the "Modified" MMAC algorithm for which the filter and controller were well tuned [26: Chapter 1 pp 13-15]. The ME/I approach and the "Modified" MMAC algorithm will be fully discussed in Chapter 2.

The most recent work is that of Gustafson [2, 3], who applied the current research to the SPICE 2 model. This was the first time this line of research was applied to a total, actual system. Gustafson's work showed good performance of the moving-bank MMAE/MMAC with a 134-state filter model versus a 194-state truth model. However, with more stringently reduced order filter models, the MMAC could not quell the disturbance vibrations adequately. Additionally, the good performance of the controller was marred by its inability to keep the LOS deviations within the 1 μ-radian specification. The inability to meet this specification was attributed to poor actuator and sensor performance and did not lie in the MMAE/MMAC adaptation process: even a nonadaptive artificially informed controller failed to meet the specification.

Gustafson also ran tests using an expanded **H** matrix which resulted in better quelling, but the specification was still not met [2: Chapter 5 pg. 3]. This expanded **H** matrix simulated relative position and velocity measurements of the sensor measurement points on the structure. These extra measurements were not physically realizable but were useful for a comparison study. The performance improvement shown here can be equated to an expected equal or greater performance gain if the sensor measurement precision is improved.

1.3.2 Dithering Applications

Much work has been done in the past using dither signals to enhance parameter identifiability. A few of the applications which utilized these techniques will be presented here.

In this direct line of research prior to Gustafson [1, 5-8, 11, 12, 22, 23, 26, 29, 32], dithering signals were input to the system in order to obtain an adequate uncertain parameter identification; there were no specifically *designed* dithers, just some rather arbitrarily chosen excitation signals. This was necessary because the dynamics driving noise was too small for the parameters to be identified without some form of forced excitation. The system needed to be "shaken up" with a dither signal in order for any parameter identification to take place. Typically, square wave dither signals were used in this previous work, but with no purposeful structure. However, with the SPICE 2 model used by Gustafson, a sufficient magnitude of dynamics driving noise was present to operate somewhat successfully in open loop without the use of a dither signal, so none was applied. Gustafson showed that the MMAE was somewhat able to track a varying parameter under these conditions [2: Chapter 5 pp. 46-49].

Menke and Maybeck [24, 25] used a variety of dither input signals in an MMAC application for detecting sensor and/or actuator failures in a flight control system: sine wave, square wave, triangle wave, and pulse trains, as well as purposeful maneuver commands. All of these dither inputs performed well, but the sine wave gave the best performance [24: 203].

Hanlon [4: 36-37] looked at the eigenvalues in a flight control system to determine the best sine wave dither signal to use for enhancing failure detection. The eigenvalues provided direct insight as to which dither frequency to use. Before using the insight gained from the eigenvalues, an arbitrarily chosen dither frequency was chosen which did not produce useful results.

1.4 Objectives

The purpose of this thesis is threefold. First, various dithering techniques will be implemented in order to enhance uncertain parameter identifiability for the SPICE 2 model in open-loop estimation prior to closed-loop control. It is envisioned that this will provide enhanced initial parameter estimates when the closed-loop control system is turned on; there is enough excitation in the closed-loop phase that the dither would only be used in the prior open-loop phase. The research will also explore whether or not naturally occurring slew commands will excite the system sufficiently without dithering. Second, increased sensor measurement precision will be investigated in an effort to increase controller performance (decrease LOS deviations). It is anticipated that reduced-order design models will be more fruitful once measurement precision is improved. Third, an alternative to Gustafson's discretized parameter space will be developed in an order to improve MMAE/MMAC overall performance.

1.5 Scope

Since this thesis research is a direct continuation of Gustafson's [2], this research will utilize the existing SPICE 2 model in addition to the Kalman filter and controller tuning developed by Gustafson, except where modifications are necessary (see next section). The single unknown parameter (undamped natural frequency of the structural bending modes) will continued to be used in this research. The same discretized parameter space developed by Gustafson will also be used, except when a new discretization is formed.

The limits placed on the modeling are as follows. The truth model is a linearized model for the SPICE 2 structure. The Kalman filter is based on a reduced-order linear time-invariant model, driven by stationary noise. The LQG controller is developed from constant state and control weighting matrices with no cross-weighting matrix. Gain transients in both the filter and controller will be ignored, in order to exploit the steady-state constant-gain Kalman filter in the MMAE, and the steady-state constant-gain LQG controller in the MMAC.

1.6 Approach

To apply the dithering techniques, two items will be accomplished. First, several forms of dither signal inputs will be considered in order to observe the effects each has on enhancing parameter identification. The different inputs to be considered are: (1) square wave, (2) sine wave, (3) swept frequency sine wave, (4) wide-band noise (limited to the expected band of undamped natural frequencies). Second, rigid-body states will be augmented to the SPICE 2 structural model to simulate a commanded rotation or slewing motion of the structure. The effects of this rigid-body motion will be observed with respect to how it affects parameter identification through excitation of the bending modes, just as with the dither signal inputs. Third, simulations will be run with the sensor measurement precision increased by one, two or more orders of magnitude. With an increase in the sensor measurement precision (decrease in sensor noise), the effects in the MMAC performance will be observed when based on more stringently reduced-order models (with associated decrease of computational loading).

With the dithering inputs, a comparison will be accomplished among the various techniques to determine which, if any, is "best". Minor modifications to the existing software [2] will be required.

Increasing the sensor measurement precision will require retuning of the existing Kalman filter designs (assumed dynamics noise strength, $\mathbf{Q}$, and assumed measurement noise covariance, $\mathbf{R}$). Before changing the precision by one or two orders of magnitude, the effects of perfect measurements (no measurement noise) will be determined. In this way, a limiting case in the MMAC performance will be known, with regard to increased sensor measurement precision.

The change to the parameter space discretization will be either to change the location of discrete point values, or to allow a different number of points, or both. The existing span of values for the undamped natural frequencies will be maintained, however, for the purpose of making an accurate comparison of the MMAE/MMAC performance with the old and new parameter space discretization. Minor software modifications to the existing software will be required here as well.

1.7 Summary

A general introduction and background in the area of utilizing the Kalman filter and LQG controller with a multiple model approach to achieving desired performance despite parameter uncertainty were presented in this chapter. The techniques to be applied toward this research were also presented, namely: dithering techniques for uncertain parameter open-loop identification enhancement, increased sensor measurement precision for improved MMAE/MMAC performance, and modified parameter discretization for better parameter tracking performance.

To motivate the desire for the use of the multiple model approach and the dithering techniques to be implemented, a summary of previous research in these areas was also presented. The specific objectives of this research along with a defined scope and general approach (methodology) were then presented.

Chapter 2 will develop the fundamental concepts of the Kalman filter and LQG controller. These concepts will then be used to develop the moving-bank MMAE and MMAC algorithms. Chapter 2 will also present a section on applying dithering input signals and a section on mathematical modeling of the physical structure. Chapter 3 will present a summary of the SPICE 2 structure model developed by Gustafson [2: Chapter 3]. Additionally, Chapter 3 will include the development for augmenting rigid-body states to the SPICE 2 model. The software requirements and simulations to be performed will be presented in Chapter 4. Chapter 5 will include a presentation of the results found in this research. Chapter 6 will finish with a discussion of the conclusions drawn from the results, along with recommendations for future research.

2. Development of Fundamental Concepts

2.1 Introduction

Three main concepts will be developed in this chapter: the Kalman filter, the state and parameter estimator, and the stochastic controller. The Kalman filter development will be thorough but basic. For the state and parameter estimator, Multiple Model Adaptive Estimation (MMAE) algorithms will be implemented, and this chapter will present different methods of implementing the MMAE. The stochastic controller will utilize Linear-Quadratic-Gaussian (LQG) techniques as part of the Multiple Model Adaptive Controller (MMAC). As with the MMAE, various methods of implementing the MMAC will be addressed.

In addition to these three main concepts, two additional items will be discussed. First, algorithms for implementing the dithering techniques for enhanced parameter identification will be developed. Second, a mathematical modeling section will be presented in which physical and modal forms are developed in order to demonstrate the modal reduction technique of producing a reduced-order filter model from a truth model.

The development presented in this chapter is independent of the particular system of concern, e.g., a flexible space-structure. For this reason, the equations and algorithms developed here can be applied to almost any situation which is constrained by the same assumptions contained in this work.

2.2 Kalman Filter (Bayesian Estimation)

In order to utilize the simplest form of the Kalman filter, the system under study is assumed adequately modeled by a continuous-time, stochastic linear differential equation driven by zero-mean white Gaussian noise. The sampled-data measurement model has similar assumptions, i.e., linear measurements disturbed by zero-mean white Gaussian noise. Thus, the Kalman filter is based on the following linear stochastic differential equation:

$$\dot{\mathbf{x}}(t) = \mathbf{F}(t)\mathbf{x}(t) + \mathbf{B}(t)\mathbf{u}(t) + \mathbf{G}(t)\mathbf{w}(t) \tag{2.1}$$

where $\mathbf{x}(t)$ is the n-dimensional state vector, $\mathbf{u}(t)$ is the r-dimensional deterministic control input vector, and $\mathbf{w}(t)$ is the s-dimensional white Gaussian noise vector. $\mathbf{F}(t)$ is the n-by-n dimensional system dynamics matrix, $\mathbf{B}(t)$ is the n-by-r dimensional deterministic input matrix, and $\mathbf{G}(t)$ is the n-by-s dimensional noise input matrix.

As indicated above, the driving noise has statistics of:

$$E\{\mathbf{w}(t)\} = \mathbf{0} \tag{2.2}$$

$$E\left\{\mathbf{w}(t)\mathbf{w}(t')^{\mathrm{T}}\right\} = \mathbf{Q}(t)\delta(t-t') \tag{2.3}$$

where $\mathbf{Q}(t)$ is an s-by-s dimensional, symmetric, positive semi-definite matrix and indicates the strength of the dynamics driving noise; $\delta(t)$ is the Dirac delta function.

As with all stochastic linear differential equations, initial conditions must be specified. These are in the form of $\hat{\mathbf{x}}_0$ and $\mathbf{P}_0$. With these two initial conditions known, the entire corresponding Gaussian probability density function can be described. The initial conditions are as follows:

$$E\{\mathbf{x}(t_0)\} = \hat{\mathbf{x}}_0 \tag{2.4}$$

$$E\left\{[\mathbf{x}(t_0) - \hat{\mathbf{x}}_0][\mathbf{x}(t_0) - \hat{\mathbf{x}}_0]^{\mathrm{T}}\right\} = \mathbf{P}_0 \tag{2.5}$$

Here, $\hat{\mathbf{x}}_0$ is the mean or "best guess" of the initial state vector. $\mathbf{P}_0$ is the initial covariance or indicator of the "spread" of the density function.

With the ultimate implementation of the Kalman filter on a digital computer, we need a discretized set of Kalman filter equations. It is well known that, in general, it is preferable to develop a discrete-time filter around an equivalent discrete-time system model, rather than discretizing a continuous-time filter [19:261]. Thus, the discretized system model is given as:

$$\mathbf{x}(t_i) = \mathbf{\Phi}(t_i, t_{i-1})\mathbf{x}(t_{i-1}) + \mathbf{B}_d(t_{i-1})\mathbf{u}(t_{i-1}) + \mathbf{G}_d(t_{i-1})\mathbf{w}_d(t_{i-1}) \tag{2.6}$$

where $\mathbf{\Phi}(t_i, t_{i-1})$ is the state transition matrix associated with $\mathbf{F}(t)$, i.e., the solution to:

$$\dot{\mathbf{\Phi}}(t, t_{i-1}) = \mathbf{F}(t)\mathbf{\Phi}(t, t_{i-1}) \tag{2.7}$$

with the initial condition:

$$\mathbf{\Phi}(t_{i-1}, t_{i-1}) = \mathbf{I} \tag{2.8}$$

Assuming time invariant $\mathbf{F}$, the Laplace transform solution yields

$$\mathbf{\Phi}(t_i, t_{i-1}) = \mathbf{\Phi}(t_i - t_{i-1}) = e^{\mathbf{F}(t_i - t_{i-1})} = \mathcal{L}^{-1}\left\{[s\mathbf{I} - \mathbf{F}]^{-1}\right\}\Big|_{(t_i - t_{i-1})} \tag{2.9}$$

If we then assume constant $\mathbf{u}$ over each sample period, $\mathbf{B}_d(t_{i-1})$ is solved by:

$$\mathbf{B}_d(t_{i-1}) = \int_{t_{i-1}}^{t_i} \mathbf{\Phi}(t_i, \tau)\mathbf{B}(\tau)\,d\tau \tag{2.10}$$

Similar to the continuous time case, the statistics of the dynamics driving noise are given by:

$$E\{\mathbf{w}_{\mathrm{d}}(t_{i-1})\} = \mathbf{0} \tag{2.11}$$

$$E\{\mathbf{w}_{\mathrm{d}}(t_{i-1})\mathbf{w}_{\mathrm{d}}(t_j)^{\mathrm{T}}\} = \mathbf{Q}_{\mathrm{d}}(t_{i-1})\delta_{(i-1)j} \tag{2.12}$$

where $\delta_{(i-1)j}$ is the Kronecker delta function. $\mathbf{Q}_{\mathrm{d}}(t)$ is given by:

$$\mathbf{Q}_{\mathrm{d}}(t_{i-1}) = \int_{t_{i-1}}^{t_i} \mathbf{\Phi}(t_i,\tau)\mathbf{G}(\tau)\mathbf{Q}(\tau)\mathbf{G}^{\mathrm{T}}(\tau)\mathbf{\Phi}^{\mathrm{T}}(t_i,\tau)d\tau \tag{2.13}$$

Measurements are obtained by the discrete-time measurement model:

$$\mathbf{z}(t_i) = \mathbf{H}(t_i)\mathbf{x}(t_i) + \mathbf{v}(t_i) \tag{2.14}$$

where $\mathbf{z}(t_i)$ is an m-dimensional discrete-time measurement vector. $\mathbf{H}(t_i)$, is an m-by-n dimensional measurement matrix and $\mathbf{v}(t_i)$, is an m-dimensional noise vector representing the uncertainty in the measurement information. As with the dynamics driving noise, $\mathbf{w}_{\mathrm{d}}(t_{i-1})$ in Equation (2.6), this vector is modeled as white, Gaussian noise with statistics given by:

$$E\{\mathbf{v}(t_i)\} = \mathbf{0} \tag{2.15}$$

$$E\{\mathbf{v}(t_i)\mathbf{v}(t_j)^{\mathrm{T}}\} = \mathbf{R}(t_i)\delta_{ij} \tag{2.16}$$

where $\mathbf{R}(t_i)$ is an m-by-m dimensional, positive definite, symmetric matrix and δ_{ij} is the Kronecker delta function. $\mathbf{R}(t_i)$ being positive definite indicates that each measurement component is noise-corrupted and that no linear combination of these components is noise-free.

One final assumption is that $\mathbf{x}(t_0)$, $\mathbf{w}(t)$ and $\mathbf{v}(t_i)$ are assumed independent and thus uncorrelated with each other. The equivalence of independence and uncorrelatedness comes from the joint Gaussian assumption [19:205].

Now that the discrete-time system is described, the propagation/update cycle of the Kalman filter will be addressed. The propagation portion of this cycle consists of propagating (moving ahead in time) the most recent state estimate information at time, t_{i-1}^+, to time, t_i^-. The superscript "+" denotes the time just after a measurement update, while the superscript "-" denotes the time just prior to a measurement update. Just after the propagation comes a measurement update. Thus, this cycle can be described by the following sequence:

$$t_0^+, \ldots t_1^-, t_1^+, \ldots t_2^-, t_2^+, \ldots\ldots, t_i^-, t_i^+, \ldots t_{i+1}^-, \ldots$$

The propagation equations are given by:

$$\hat{\mathbf{x}}(t_i^-) = \mathbf{\Phi}(t_i, t_{i-1})\hat{\mathbf{x}}(t_{i-1}^+) + \mathbf{B}_d(t_{i-1})\mathbf{u}(t_{i-1}) \tag{2.17}$$

$$\mathbf{P}(t_i^-) = \mathbf{\Phi}(t_i, t_{i-1})\mathbf{P}(t_{i-1}^+)\mathbf{\Phi}^T(t_i, t_{i-1}) + \mathbf{G}_d(t_{i-1})\mathbf{Q}_d(t_{i-1})\mathbf{G}_d^T(t_{i-1}) \tag{2.18}$$

The following set of update equations are used to update the state estimate with the information provided by the current measurement:

$$\mathbf{K}(t_i) = \mathbf{P}(t_i^-)\mathbf{H}^T(t_i)\left[\mathbf{H}(t_i)\mathbf{P}(t_i^-)\mathbf{H}^T(t_i) + \mathbf{R}(t_i)\right]^{-1} \tag{2.19}$$

$$\hat{\mathbf{x}}(t_i^+) = \hat{\mathbf{x}}(t_i^-) + \mathbf{K}(t_i)\left[\mathbf{z}_i - \mathbf{H}(t_i)\hat{\mathbf{x}}(t_i^-)\right] \tag{2.20}$$

$$\mathbf{P}(t_i^+) = \mathbf{P}(t_i^-) - \mathbf{K}(t_i)\mathbf{H}(t_i)\mathbf{P}(t_i^-) \tag{2.21}$$

where $\mathbf{z}_i$ is the measurement to be incorporated at time t_i (a specific realization of $\mathbf{z}(t_i)$), and $\mathbf{K}(t_i)$ is the Kalman filter gain matrix. The residual equation is given by:

$$\mathbf{r}(t_i) = \mathbf{z}_i - \mathbf{H}(t_i)\hat{\mathbf{x}}(t_i^-) \tag{2.22}$$

where $\mathbf{r}(t_i)$ is termed the residual and is simply the bracketed quantity in Equation (2.20) [10:217-218]. The residual term is singled out because of its special importance in the system analysis and the MMAE to follow in the next section. The residual is seen to be the difference between the current measurement and the filter's best "guess" of what that measurement should be. This difference "guides" the filter in the correct direction (+ or -) for updating the state estimate. The magnitude of the update is controlled by the gain matrix. The filter anticipates that the residual will be Gaussian, of mean zero and of covariance

$$\mathbf{A}(t_i) = \mathbf{H}(t_i)\mathbf{P}(t_i^-)\mathbf{H}^{\mathrm{T}}(t_i) + \mathbf{R}(t_i) \tag{2.23}$$

which is inherently computed by the filter in the gain calculation of Equation (2.19). Since $\hat{\mathbf{x}}(t_i^+)$ is formed from $\hat{\mathbf{x}}(t_i^-)$ in Equation (2.20), the state estimate always incorporates knowledge (information) of previous measurements. This is the underlying idea behind recursive Bayesian estimation.

With the Kalman filter system of equations presented, we will discuss some further simplifying assumptions which will minimize the computational work involved with implementing such a system. The following simplifications revolve around the current assumptions of a time-invariant system with stationary noise processes. With this type of system the filter is known to have a transient followed by a steady state response in $\mathbf{P}$ and $\mathbf{K}$ [19:224]. If the transient period is relatively short, relative to the total time of interest, then the steady state values may be used (if performance degradation is sufficiently small). Thus, with the time-invariant assumption and a fixed sample period, $\mathbf{P}^-$, $\mathbf{P}^+$ and $\mathbf{K}$ may be precomputed once for all time, as can $\mathbf{B}_d$, and relieve a considerable computational burden. With the addition of the stationary noise assumption, $\mathbf{Q}_d$ can be precomputed once for all time as well. For the system under consideration, another valid assumption is for $\mathbf{R}$ to be constant as well.

With the simplifying assumptions in place, the last step in the Kalman filter system development is that of "tuning". This process involves adjusting or "tuning" the values for $\hat{\mathbf{x}}_0$, $\mathbf{R}$ and $\mathbf{Q}_d$. The development of these and the other system model matrices (choice of state variables in $\mathbf{x}$; $\mathbf{F}$, $\mathbf{\Phi}$, $\mathbf{B}_d$, $\mathbf{u}$, $\mathbf{G}_d$ and $\mathbf{H}$) will be presented in Chapter 3. The actual tuning process will be presented in Chapter 4.

2.3 MMAE

The basic understanding of the *Multiple Model Adaptive* Estimator (MMAE) was discussed in Chapter 1. This section will expand on this idea, starting with the uncertain parameter problem and the Bayesian formulation of the MMAE. This is followed by a discussion on underlying problems with the basic or raw form of the Bayesian MMAE. This section is concluded with a discussion on proposed enhancements to address these problems.

2.3.1 Bayesian MMAE

Uncertainties are already a given with a stochastic system (dynamics and measurement noise), but these noises are white stationary processes. The unknown or uncertain parameter comes into play when forming a system matrix ($\mathbf{\Phi}$ in our case). Note, that in general, uncertain parameters can affect $\mathbf{B}_d, \mathbf{H}, \mathbf{Q}_d$ and/or $\mathbf{R}$ as well. The Kalman filter for a particular system is only as accurate as the system matrices which make up the filter. If there exist some uncertainty in the system at hand (damping ratio and undamped natural frequency of various structural bending modes in our case), one method of handling this is the MMAE.

To understand the Bayesian formulation of the MMAE, consider the following. Let $\mathbf{a}$ be the random vector composed of the uncertain parameters. In this initial discussion, $\mathbf{a}$ will be considered a time-invariant random vector. As presented in Chapter 1, the parameter space will be comprised of a set of K parameter vector values, $\{\mathbf{a}_1, \mathbf{a}_2, \ldots \mathbf{a}_k, \ldots \mathbf{a}_K\}$. This discretized space is preferred rather than a continuous space for obvious dimensionality problems; later it will be seen to allow an MMAE implementation with a finite number, rather than an infinite number, of elemental Kalman filters. (The actual parameter space discretization will be developed in Chapter 3.) Now consider a joint density function of $\mathbf{x}(t_i)$ and $\mathbf{a}$, conditioned on measurements $\mathbf{Z}(t_i)$. Here, $\mathbf{Z}(t_i)$ is the measurement history through time t_i, made up of the components $\mathbf{z}(t_1)$, $\mathbf{z}(t_2)$, ..., $\mathbf{z}(t_i)$. Maybeck [20:76,129] states this is the logical choice for a Bayesian formulation and is given by:

$$f_{\mathbf{x}(t_i),\mathbf{a}|\mathbf{Z}(t_i)}(\xi,\alpha|\mathbf{Z}_i) = f_{\mathbf{x}(t_i)|\mathbf{a},\mathbf{Z}(t_i)}(\xi|\alpha,\mathbf{Z}_i)\, f_{\mathbf{a}|\mathbf{Z}(t_i)}(\alpha|\mathbf{Z}_i) \tag{2.24}$$

Note that Bayes' rule [19:81] is applied to obtain the equality. This form of the density function is useful since the two terms on the right of Equation (2.24) are readily known. The first term on the right is the Gaussian density defined by the Kalman filter based on the assumption that $\mathbf{a}$ assumes the value α, with mean $\hat{\mathbf{x}}(t_i^+)$ and covariance $\mathbf{P}(t_i^+)$ [20:129]. The second density function in Equation (2.24) is formed by considering the probability of $\mathbf{a}$ taking on the value α at time t_i. The density function is then described by [20:130]:

$$f_{\mathbf{a}|\mathbf{Z}(t_i)}(\boldsymbol{\alpha}|\mathbf{Z}_i) = \sum_{k=1}^{K} p_k(t_i)\delta(\boldsymbol{\alpha} - \mathbf{a}_k) \tag{2.25}$$

Then, for any time t_i, the conditional hypothesis probability is recursively defined as [20:131]:

$$\begin{aligned} p_k(t_i) &\equiv \text{prob}\{\mathbf{a} = \mathbf{a}_k | \mathbf{Z}(t_i) = \mathbf{Z}_i\} \\ &= \frac{f_{\mathbf{z}(t_i)|\mathbf{a},\mathbf{Z}(t_{i-1})}(\mathbf{z}_i \,|\, \mathbf{a}_k, \mathbf{Z}_{i-1})\, p_k(t_{i-1})}{\sum_{j=1}^{K} f_{\mathbf{z}(t_i)|\mathbf{a},\mathbf{Z}(t_{i-1})}(\mathbf{z}_i \,|\, \mathbf{a}_j, \mathbf{Z}_{i-1})\, p_j(t_{i-1})} \end{aligned} \tag{2.26}$$

where each measurement conditional density is evaluated as [20:132]:

$$f_{\mathbf{z}(t_i)|\mathbf{a},\mathbf{Z}(t_{i-1})}(\mathbf{z}_i|\mathbf{a}_k, \mathbf{Z}_{i-1}) = \frac{1}{(2\pi)^{m/2}|\mathbf{A}_k(t_i)|^{1/2}} \exp\left[-\tfrac{1}{2}\mathbf{r}_k^{\mathrm{T}}(t_i)\mathbf{A}_k^{-1}(t_i)\mathbf{r}_k(t_i)\right] \tag{2.27}$$

where $\mathbf{r}_k(t_i)$ and $\mathbf{A}_k(t_i)$ are computed in the kth Kalman filter (based on the assumption that $\mathbf{a} = \mathbf{a}_k$) as:

$$\mathbf{r}_k(t_i) = \mathbf{z}(t_i) - \mathbf{H}_k(t_i)\hat{\mathbf{x}}_k(t_i^-) \tag{2.28}$$

and

$$\mathbf{A}_k(t_i) = \mathbf{H}_k(t_i)\mathbf{P}_k(t_i^-)\mathbf{H}_k^{\mathrm{T}}(t_i) + \mathbf{R}_k(t_i) \tag{2.29}$$

From this Bayesian formulation we will have a state estimate (conditional mean) and covariance, both of which are conditioned on the past measurement history. The conditional mean is formed by [20:131]:

$$\begin{aligned}
\hat{\mathbf{x}}(t_i^+) &= E\{\mathbf{x}(t_i) | \mathbf{Z}(t_i) = \mathbf{Z}_i\} \\
&= \int_{-\infty}^{\infty} \xi \left[\sum_{k=1}^{K} f_{\mathbf{x}(t_i)|\mathbf{a},\mathbf{Z}(t_i)}(\xi | \mathbf{a}_k, \mathbf{Z}_i) p_k(t_i) \right] d\xi \\
&= \sum_{k=1}^{K} \hat{\mathbf{x}}_k(t_i^+) p_k(t_i)
\end{aligned} \tag{2.30}$$

where $\hat{\mathbf{x}}_k(t_i^+)$ is produced by the *k*th Kalman filter based on the parameter $\mathbf{a}_k$. Note the difference between $\hat{\mathbf{x}}(t_i^+)$ and $\hat{\mathbf{x}}_k(t_i^+)$. $\hat{\mathbf{x}}(t_i^+)$ is merely the probabilistically weighted average of each of the Kalman filter state estimates. Refer back to Figure 1-1 for a graphical depiction of this process, also known as the *multiple model filtering process* [20:131].

If it is desired to obtain the conditional covariance of $\mathbf{x}(t_i)$, it can be found from [20:131]:

$$\begin{aligned}
\mathbf{P}(t_i^+) &= E\left\{ [\mathbf{x}(t_i) - \hat{\mathbf{x}}(t_i^+)][\mathbf{x}(t_i) - \hat{\mathbf{x}}(t_i^+)]^{\mathrm{T}} | \mathbf{Z}(t_i) = \mathbf{Z}_i \right\} \\
&= \int_{-\infty}^{\infty} [\xi - \hat{\mathbf{x}}(t_i^+)][\xi - \hat{\mathbf{x}}(t_i^+)]^{\mathrm{T}} f_{\mathbf{x}(t_i)|\mathbf{Z}(t_i)}(\xi | \mathbf{Z}_i) d\zeta \\
&= \sum_{k=1}^{K} p_k(t_i) \left\{ \int_{-\infty}^{\infty} [\xi - \hat{\mathbf{x}}(t_i^+)][\xi - \hat{\mathbf{x}}(t_i^+)]^{\mathrm{T}} f_{\mathbf{x}(t_i)|\mathbf{a},\mathbf{Z}(t_i)}(\xi | \mathbf{a}_k, \mathbf{Z}_i) d\xi \right\} \\
&= \sum_{k=1}^{K} p_k(t_i) \left\{ \mathbf{P}_k(t_i^+) + [\hat{\mathbf{x}}_k(t_i^+) - \hat{\mathbf{x}}(t_i^+)][\hat{\mathbf{x}}_k(t_i^+) - \hat{\mathbf{x}}(t_i^+)]^{\mathrm{T}} \right\}
\end{aligned} \tag{2.31}$$

where $\mathbf{P}_k(t_i^+)$ is the covariance of each Kalman filter's state estimate based on the parameter $\mathbf{a}_k$, as explicitly precomputable in the kth Kalman filter. Note here that $\mathbf{P}(t_i^+)$ cannot be precomputed since it is directly tied to knowledge of the measurement history.

In order to evaluate the performance of the multiple model parameter estimation, it will be useful to calculate the conditional mean and covariance of the parameter. These are given by [20:132-133]:

$$\begin{aligned}\hat{\mathbf{a}}(t_i) \equiv E\{\mathbf{a}(t_i)|\mathbf{Z}(t_i)=\mathbf{Z}_i\} &= \int_{-\infty}^{\infty} \boldsymbol{\alpha} f_{\mathbf{a}|\mathbf{Z}(t_i)}(\boldsymbol{\alpha}|\mathbf{Z}_i)\,d\boldsymbol{\alpha} \\ &= \int_{-\infty}^{\infty} \boldsymbol{\alpha}\left[\sum_{k=1}^{K} p_k(t_i)\,\delta(\boldsymbol{\alpha}-\mathbf{a}_k)\right] d\boldsymbol{\alpha} \\ &= \sum_{k=1}^{K} \mathbf{a}_k p_k(t_i) \end{aligned} \tag{2.32}$$

and:

$$\begin{aligned}\mathbf{P}_{\hat{\mathbf{a}}}(t_i) &= E\left\{[\mathbf{a}-\hat{\mathbf{a}}(t_i)][\mathbf{a}-\hat{\mathbf{a}}(t_i)]^{\mathrm{T}}|\mathbf{Z}(t_i)=\mathbf{Z}_i\right\} \\ &= \sum_{k=1}^{K} [\mathbf{a}_k-\hat{\mathbf{a}}(t_i)][\mathbf{a}_k-\hat{\mathbf{a}}(t_i)]^{\mathrm{T}} p_k(t_i) \end{aligned} \tag{2.33}$$

2.3.2 Enhancements to Bayesian Estimation

The preceding section gave the Bayesian formulation in multiple model form for the problem of state estimation (and parameter estimation) in the face of uncertain parameters. This section will discuss alternate formulations of the multiple model's state estimate and other enhancements to include in the multiple model algorithm.

The Bayesian setup of the multiple model adaptive estimation algorithm in its pure form has two potential drawbacks. For each, there exist modifications or enhancements to overcome these drawbacks. In order to discuss the first drawback, consider the conditional probability density function in Equation (2.27). Normally the density function with the largest magnitude is the one which contains the smallest residual term, $\mathbf{r}_k$. This in turn produces the highest probability for the *k*th Kalman filter in Equation (2.26). This is how the algorithm should work, since the filter with the smallest residual is likely to be based on a model that is closest to the truth model. However, if the situation arises in which each $\mathbf{r}_k^{\mathrm{T}}\mathbf{A}^{-1}\mathbf{r}_k$ is of the same relative magnitude (which would be desired to yield equal probabilities for each elemental Kalman filter), then the magnitude of the conditional density function will be driven by the $\frac{1}{|\mathbf{A}_k|}$ term in Equation (2.27). In this case the filter with the smallest $|\mathbf{A}_k|$ will produce the largest density value and thus the largest probability. This is certainly not the criterion upon which to base the probabilistically weighted state estimate. To overcome this possible shortcoming of the pure Bayesian formulation, an alternative or enhancement is available and is discussed below.

For this enhancement we will assume that the residuals are Gaussian with an identity covariance matrix. This implies that the residuals will follow a "maximally noncommittal residual distribution" [10:26]. The enhancement is termed *Maximum Entropy with Identity Covariance* (ME/I). With this method the $\mathbf{A}_k$ term in the conditional density function of Equation (2.27) is replaced with the identity matrix, $\mathbf{I}$. The resulting density function is given by:

$$f_{\mathbf{z}(t_i)|\mathbf{a},\mathbf{Z}(t_{i-1})}(\mathbf{z}_i|\mathbf{a}_k,\mathbf{Z}_{i-1}) = \frac{1}{(2\pi)^{m/2}}\exp\left[-\tfrac{1}{2}\mathbf{r}_k^{\mathrm{T}}(t_i)\mathbf{r}_k(t_i)\right] \tag{2.34}$$

Note that, without this modification, we assume the residuals are Gaussian with covariance $\mathbf{A}_k$ and that "a Gaussian distribution is the maximum entropy distribution" [29:32, 10:24]. Thus, the method of using the conditional density function with covariance $\mathbf{A}_k$ (Equation 2.27) is termed *Maximum Entropy with Covariance* $\mathbf{A}_k$ (ME/A).

A second problem can occur if a probability, $p_k(t_i)$, is ever calculated to be zero. Because of the iterative nature of the probability calculation in Equation (2.26), if $p_k(t_i)$ ever becomes zero it will remain at zero for all time thereafter. This condition is often termed "lock-out." To prevent such a condition, an ad hoc technique of "lower bounding" $p_k(t_i)$ is used [20:135, 17]. This enhancement of setting a lower bound, p_{min}, on $p_k(t_i)$ prevents the probability from ever becoming zero. The lower bound is set sufficiently low so as to not bias the computation, and at each sample time, t_i, after each probability is computed with Equation (2.26), each $p_k(t_i)$ is rescaled such that $\sum_{k=1}^{K} p_k(t_i) = 1$.

In addition to the enhancements presented above, there is another method of computing the state and parameter estimate. This method is termed *Maximum A Posteriori* (MAP) and assumes the filter with the highest associated probability will provide the best state estimate. This method can be used with or without the ME/I technique of computing the probability density function.

2.4 Moving-Bank MMAE Development

The moving-bank concept was introduced in Chapter 1 and is an enhancement to fixed-bank (full-bank) MMAE. In order for the MMAE to be robust, the parameter space must be "large", but this, of course, can create a burden for the microprocessor if the discrete point values are finely distributed throughout the space. To alleviate this burden, a subset of J filters is used at any given time (with $J < K$). Then, for the MMAE to be effective, the subset of filters must "move" through the parameter space (to keep up with changes in true parameter value, $\mathbf{a}$). In order for the move to work properly, the bank will also have to be expanded and contracted periodically. Additionally, as the bank moves through the parameter space, new elemental Kalman filters will have to be initialized. In this section, techniques for moving, expanding and contracting the bank, as well as initializing filters, will be discussed.

2.4.1 Moving the Bank

As stated previously, the moving-bank MMAE is the J subset of filters which surround the most likely correct filter. When parameter variations take place, the bank will have to move to keep itself centered over the correct filter. This moving-bank concept is depicted in Figure 1-6. In order to accomplish the bank move, three methods (or logics) are presented. They are (1) Residual Monitoring, (2) Parameter Position Estimate Monitoring, and (3) Probability Monitoring.

2.4.1.1 Residual Monitoring

The residual monitoring method makes use of the likelihood quotient, $L_j(t_i)$, which is defined as:

$$L_j(t_i) = \mathbf{r}_j^{\mathrm{T}}(t_i)\mathbf{A}_j(t_i)^{-1}\mathbf{r}_j(t_i) \tag{2.35}$$

Note that the likelihood quotient has the same quadratic form as the exponential term in Equation (2.27). Thus at each sample time, $L_j(t_i)$ is computed for each of the J filters. Each of the J likelihood quotient values is compared to a set threshold. Computed values larger than the threshold are an indication the bank needs to be moved (in the direction of the smaller values). If all $L_j(t_i)$ values are greater than a set threshold, then the bank can be moved in the direction of the smallest value, or the bank can be expanded (to be explained shortly). The direction of movement (determined from $L_j(t_i)$ with the smallest value) should correspond to the parameter $\mathbf{a}_j$ being closest to the true parameter. Since this method relies on the assumption that the most likely true filter location is the one corresponding to the smallest current residual or in this case the smallest current likelihood quotient value, then it is inherently troubled with possible single large residual values giving an incorrect indication of the move direction. For this reason, the move decision threshold must be set high enough so the bank will not move in an erratic fashion [22:1876].

2.4.1.2 Parameter Position Estimate Monitoring

This method uses the parameter estimate itself to keep the moving-bank centered over the most likely correct filter model. This is accomplished by comparing the current parameter estimate from Equation (2.32) to the bank's center. If the difference is greater than a set threshold, then the bank can be moved in the appropriate direction so as to put its center as close as possible to the current parameter estimate.

In comparing this method with the previous one, it is seen that using the parameter position estimate is a more direct interpretation of the information indicating the true location of the uncertain parameter. It is also seen that this method utilizes the entire past measurement history, whereas the Residual Monitoring method utilizes only the current measurement. Thus, the Parameter Position Estimate Monitoring method does not suffer from possible erratic behavior due to single large residual values.

2.4.1.3 Probability Monitoring

This method combines properties of the previous two methods and attempts to center the bank over the most likely filter model by utilizing the conditional probabilities determined from Equation (2.26). The $p_j(t_i)$ with the highest value (greater than a set threshold) is used to determine the bank's center, and the bank will move in the direction of the filter with the highest computed probability. Thus, (as with Residual Monitoring) this method uses residual information, except that here the entire measurement history is utilized as well. Then, (as with Position Monitoring) this method is not susceptible to erratic behavior due to single large residual values.

2.4.2 Expanding the Bank

When parameter variations are "small", then the set of filters can move as a finely discretized bank. If, however, the true parameter makes a large jump in value in the parameter space, the bank may need to be expanded to a coarse discretization in order to encompass the true parameter within the bank. The decision logic of whether or not to expand the bank is based on the likelihood quotient Equation (2.35) (the other methods rely on $\hat{\mathbf{a}}(t_i)$ and $p_j(t_i)$, both of which are confined to the nearest neighbors to the current bank center, using fine discretization). As with Residual Monitoring, if all $L_j(t_i)$ values exceed a set threshold, the bank is expanded. This too can be prone to erratic expansions due to single large residual values. The threshold must be determined through trial and error for a given system.

In addition, the bank can be initially expanded in an effort to approximate the location of the true parameter within the entire parameter space. Once this acquisition cycle yields an estimate, $\hat{\mathbf{a}}$, the bank can be contracted to fine discretization, centered on $\hat{\mathbf{a}}$. Maybeck and Hentz [5:26, 22:92] stated that an initial coarse discretization may provide a better initial transient in parameter estimation than an initial fine discretization.

2.4.3 Contracting the Bank

After the bank has been expanded, it will need to be contracted once an accurate estimate of the uncertain parameter is obtained. The decision to contract the bank is generally controlled by a method called the *Parameter Estimation Error Covariance Monitoring* technique. The covariance of the uncertain parameter estimate, $\mathbf{P}_{\hat{\mathbf{a}}}(t_i)$, is given by Equation (2.33). Similar to previous decision making methods, when some scalar length variable associated with this covariance falls below a set threshold, the coarsely discretized bank is contracted to a finely discretized bank. Also, as with previous methods, the threshold must not be set too "high" or too "low." Hentz [5:69] found that too high a setting causes the bank to contract prematurely (without an adequate parameter estimate), and too low a setting delays the bank from contracting at the proper time.

Since $\mathbf{P}_{\hat{\mathbf{a}}}(t_i)$ is not in general a scalar, in the past there have different methods for utilizing this covariance information. Hentz used either the largest diagonal element [5:64] or the norm of $\mathbf{P}_{\hat{\mathbf{a}}}(t_i)$ [22:92], whereas Karnick proposed another method involving the probability of each side of the bank [7:28-29]. The probability of each side of the bank is given by

$$p_{side}(t_i) = \frac{\sum_{side} f_{\mathbf{z}(t_i)|\mathbf{a},\mathbf{Z}(t_{i-1})}(\mathbf{z}_i \mid \mathbf{a}_j \, , \mathbf{Z}_{i-1})}{\sum_{4\ sides} f_{\mathbf{z}(t_i)|\mathbf{a},\mathbf{Z}(t_{i-1})}(\mathbf{z}_i \mid \mathbf{a}_j, \mathbf{Z}_{i-1})} \tag{2.36}$$

With this method, if the probability of a side falls below a given threshold, that side is contracted "inward" toward the parameter estimate. However, if probability of a side goes above another threshold, that side is left alone, and the other three sides are contracted inward. The third case for this method is, if the sum of all four sides are below a set threshold, then they are all contracted inward. Thus, contracting the bank can be performed based on different methods just as with the bank move logic.

2.4.4 Initialization of New Elemental filters

An initialization process must take place when the bank is moved, expanded or contracted, or first started up. There are three steps to this process. First, the Kalman filter matrices, $\mathbf{\Phi}_k$, $\mathbf{B}_{dk}$, $\mathbf{G}_{dk}$, $\mathbf{H}_k$, $\mathbf{A}_k$, $\mathbf{D}_k$ and $\mathbf{K}_k$, and the controller matrix, $\mathbf{G}_{ck}^{*}$

(for the *k*th elemental filter and controller), must be read in from memory for the filters that now comprise the bank. Each of these matrices is constant for each particular filter (based on $\mathbf{a}_k$) and thus can be stored and need not be computed on-line. Second, each filter which is brought into use must have its state estimate, $\hat{\mathbf{x}}_k(t_i)$, initialized to the current state output from the MMAE, $\hat{\mathbf{x}}(t_i)$. Third, the filter probability weights, $p_k(t_i)$, must be initialized. There are different methods used to initialize the probability weights, and these are discussed below.

When the filters are first initialized, the probability weights are equally divided among the *J* filters ($p_j(t_0) = 1/J$ for all j). This is justified since at t_0 there is no a priori information about the probabilities. This will be the same case when the bank is expanded or contracted with the same justification. However, when the bank is moved, a more appropriate probability weight distribution is warranted for the newly declared filters. Instead of giving them all equal weighting (as in the initialization), the new filters are given an equal distribution of the deactivated probabilities (the total probability of the filters that are taken off-line by making the move). The filters that are preserved during the move retain their previously computed probabilities. Other, more complicated, methods of distributing the probabilities have been proposed and researched [17:27-29], but the method of equal distribution was found to be just as effective and will therefore be used in this research.

2.5 Stochastic Controller Development

Now that the estimator portion of the system is developed, we will move on to the controller development. As indicated in Chapter 1, an LQG stochastic controller is to be implemented. For the most general case, the cost function to be minimized is defined as [21:73]:

$$J = E\left\{ \sum_{i=0}^{N} \tfrac{1}{2}\left(\begin{bmatrix} \mathbf{x}(t_i) \\ \mathbf{u}(t_i) \end{bmatrix}^{\mathrm{T}} \begin{bmatrix} \mathbf{X}(t_i) & \mathbf{S}(t_i) \\ \mathbf{S}^{\mathrm{T}}(t_i) & \mathbf{U}(t_i) \end{bmatrix} \begin{bmatrix} \mathbf{x}(t_i) \\ \mathbf{u}(t_i) \end{bmatrix} \right) + \tfrac{1}{2}\mathbf{x}^{\mathrm{T}}(t_{N+1})\mathbf{X}_{\mathrm{f}}\mathbf{x}(t_{N+1}) \right\} \tag{2.37}$$

where:

- J = scalar cost to be minimized
- $\mathbf{x}(t_i)$ = n-dimensional state vector
- $\mathbf{X}(t_i)$ = n x n-dimensional state weighting matrix
- $\mathbf{X}_{\mathrm{f}}$ = n x n-dimensional final state weighting matrix
- $\mathbf{u}(t_i)$ = r-dimensional control input vector
- $\mathbf{U}(t_i)$ = r x r-dimensional control weighting matrix
- $\mathbf{S}(t_i)$ = n x r-dimensional cross-weighting matrix
- t_N = last time a control is applied (held constant to next sample period)
- t_{N+1} = final time

Here, $\mathbf{X}(t_i)$ and $\mathbf{X}_{\mathrm{f}}$ are real, symmetric and positive semi-definite; $\mathbf{U}(t_i)$ is real, symmetric and positive definite. $\mathbf{S}(t_i)$ is chosen such that $\left[\mathbf{X}(t_i) - \mathbf{S}(t_i)\mathbf{U}^{-1}(t_i)\mathbf{S}^{\mathrm{T}}(t_i)\right]$ is positive semi-definite and such that the augmented matrix in Equation (2.37) is positive semi-definite [21:73]. Values in $\mathbf{X}(t_i)$ and $\mathbf{X}_{\mathrm{f}}$ determine the relative penalty for not controlling the states to zero, while values in $\mathbf{U}(t_i)$ determine the relative amount of penalty for applying power to the actuators to accomplish the controlling action. For diagonal $\mathbf{X}(t_i)$ and $\mathbf{X}_{\mathrm{f}}$, the larger the values in these matrices, the more importance is placed on keeping particular states at zero. Similarly, for diagonal $\mathbf{U}(t_i)$, the larger the values, the more importance is placed on keeping control deviations small on particular actuators. The values that are placed in these weighting matrices come from experimentation and observation of the system of interest. For some desired performance, these matrices must be tuned in the same fashion as $\mathbf{Q}_{\mathrm{d}}$, $\mathbf{R}$ and $\mathbf{P}_0$ in the Kalman filter.

This correspondence between estimator and controller is part of a "duality" relationship and will be discussed further shortly [21:11].

With the quadratic cost defined above, the controller to minimize that cost is given by:

$$\mathbf{u}^*(t_i) = -\mathbf{G}_c^*(t_i)\hat{\mathbf{x}}(t_i^+) \tag{2.38}$$

which is obtained using *certainty equivalence* with the deterministic case [21:70]. The controller gain, $\mathbf{G}_c^*(t_i)$, is determined by solving the backward running Riccati recursion [21:73]:

$$\mathbf{G}_c^*(t_i) = \left[\mathbf{U}(t_i) + \mathbf{B}_d^T(t_i)\mathbf{K}_c(t_{i+1})\mathbf{B}_d(t_i)\right]^{-1}\left[\mathbf{B}_d^T(t_i)\mathbf{K}_c(t_{i+1})\mathbf{\Phi}(t_{i+1},t_i) + \mathbf{S}^T(t_i)\right] \tag{2.39}$$

$$\begin{aligned}\mathbf{K}_c(t_i) = {} & \mathbf{X}(t_i) + \mathbf{\Phi}^T(t_i)\mathbf{K}_c(t_{i+1})\mathbf{\Phi}(t_{i+1},t_i) \\ & -\left[\mathbf{B}_d^T(t_i)\mathbf{K}_c(t_{i+1})\mathbf{\Phi}(t_{i+1},t_i) + \mathbf{S}^T(t_i)\right]^T \mathbf{G}_c^*(t_i)\end{aligned} \tag{2.40}$$

solved backwards from the terminal condition

$$\mathbf{K}_c(t_{N+1}) = \mathbf{X}_f \tag{2.41}$$

For a time-invariant system the control law becomes

$$\mathbf{u}^*(t_i) = -\overline{\mathbf{G}}_c^*\hat{\mathbf{x}}(t_i^+) \tag{2.42}$$

$\overline{\mathbf{G}}_c^*$ can now be solved off-line as:

$$\overline{\mathbf{G}}_{\mathrm{c}}{}^{*} = \left[\mathbf{U} + \mathbf{B}_{\mathrm{d}}{}^{\mathrm{T}}\overline{\mathbf{K}}_{\mathrm{c}}\mathbf{B}_{\mathrm{d}}\right]^{-1}\left[\mathbf{B}_{\mathrm{d}}{}^{\mathrm{T}}\overline{\mathbf{K}}_{\mathrm{c}}\mathbf{\Phi} + \mathbf{S}^{\mathrm{T}}\right] \tag{2.43}$$

where $\overline{\mathbf{K}}_{\mathrm{c}}$ is the solution to the steady state Riccati equation:

$$\overline{\mathbf{K}}_{\mathrm{c}} = \mathbf{X} + \mathbf{\Phi}^{\mathrm{T}}\overline{\mathbf{K}}_{\mathrm{c}}\mathbf{\Phi} - \left[\mathbf{B}_{\mathrm{d}}{}^{\mathrm{T}}\overline{\mathbf{K}}_{\mathrm{c}}\mathbf{\Phi} + \mathbf{S}^{\mathrm{T}}\right]^{\mathrm{T}}\left[\mathbf{U} + \mathbf{B}_{\mathrm{d}}{}^{\mathrm{T}}\overline{\mathbf{K}}_{\mathrm{c}}\mathbf{B}_{\mathrm{d}}\right]^{-1}\left[\mathbf{B}_{\mathrm{d}}{}^{\mathrm{T}}\overline{\mathbf{K}}_{\mathrm{c}}\mathbf{\Phi} + \mathbf{S}^{\mathrm{T}}\right] \tag{2.44}$$

The time-invariant condition is welcomed for high dimensional systems (for decreased computational load). This condition is physically motivated when the initial Kalman filter transients and the final controller gain transients are both negligible. This will be the case if the two transient time intervals are small compared to the time interval that control is applied.

A note is made here concerning the "duality" relationship between the Kalman filter Riccati recursion Equations (2.18), (2.19) and (2.21), and the LQG controller Riccati recursion Equations (2.39) and (2.40). This duality relationship will be shown explicitly as follows. First, we substitute Equations (2.19) and (2.21) into Equation (2.18) (note the short notation used for compactness):

$$\mathbf{P}_i^- = \mathbf{\Phi}\mathbf{P}_{i-1}^-\mathbf{\Phi}^{\mathrm{T}} - \mathbf{\Phi}\mathbf{P}_{i-1}^-\mathbf{H}^{\mathrm{T}}\left[\mathbf{H}\mathbf{P}_{i-1}^-\mathbf{H}^{\mathrm{T}} + \mathbf{R}\right]^{-1}\mathbf{H}\mathbf{P}_{i-1}^-\mathbf{\Phi}^{\mathrm{T}} + \mathbf{G}_{\mathrm{d}}\mathbf{Q}_{\mathrm{d}}\mathbf{G}_{\mathrm{d}}{}^{\mathrm{T}} \tag{2.45}$$

Then we substitute Equation (2.39) into Equation (2.40).

$$\begin{aligned}\mathbf{K}_{\mathrm{c}i} = \mathbf{X} + \mathbf{\Phi}^{\mathrm{T}}\mathbf{K}_{\mathrm{c}i+1}\mathbf{\Phi} - \left[\mathbf{B}_{\mathrm{d}}{}^{\mathrm{T}}\mathbf{K}_{\mathrm{c}i+1}\mathbf{\Phi} + \mathbf{S}^{\mathrm{T}}\right]^{\mathrm{T}} \\ \cdot \left[\mathbf{U} + \mathbf{B}_{\mathrm{d}}{}^{\mathrm{T}}\mathbf{K}_{\mathrm{c}i+1}\mathbf{B}_{\mathrm{d}}\right]^{-1}\left[\mathbf{B}_{\mathrm{d}}{}^{\mathrm{T}}\mathbf{K}_{\mathrm{c}i+1}\mathbf{\Phi} + \mathbf{S}^{\mathrm{T}}\right]\end{aligned} \tag{2.46}$$

Now, by comparing Equations (2.45) and (2.46), the following variable relationships can be made:

$$\mathbf{X} \leftrightarrow \mathbf{G}_d\mathbf{Q}_d\mathbf{G}_d^T$$
$$\mathbf{\Phi} \leftrightarrow \mathbf{\Phi}^T$$
$$\mathbf{B}_d \leftrightarrow \mathbf{H}^T$$
$$\mathbf{U} \leftrightarrow \mathbf{R}$$

With the corresponding relationships just noted, it is seen that the Kalman filter Riccati recursion is of the exact same form as the LQG Riccati recursion. This duality relationship is worth noting because, as a result of the two Riccati equations being of the same form, they can both be solved for their steady state solutions by the same routine.

One last simplification is made concerning the cross weighting matrix, $\mathbf{S}$, in the quadratic cost function, Equation (2.37). This term has been shown in previous research [11] to have negligible impact on performance and is thus neglected.

2.6 MMAC

In parallel with the Kalman filter/MMAE development is the LQG control/Multiple Model Adaptive Control (MMAC) development. Now that the control equations have been presented, the methods by which the control is implemented are presented. Just as there are several methods of implementing the state and parameter estimation algorithms, there are various methods of implementing the controller. The following six methods are presented in this section: (1) MMAC Control, (2) "Modified" MMAC Control, (3) MMAC with MAP Control, (4) Single Fixed-Gain Control, (5) Single Changeable-Gain Control, and (6) "Modified" Single Changeable-Gain Control.

2.6.1 MMAC Control

The basic structure of the MMAC is shown in Figure 1-2. This is the basis from which the other methods are derived. With the basic MMAC algorithm, an elemental controller is cascaded with each elemental Kalman filter. This structure parallels that of the MMAE in that the output of each elemental controller is probabilistically weighted and summed to provide the optimal control, i.e.,

$$\mathbf{u}^*(t_i) = \sum_{k=1}^{K} \mathbf{u}_k^*(t_i) p_k(t_i) \tag{2.47}$$

where,

$$\mathbf{u}_k^*(t_i) = -\overline{\mathbf{G}}_c^*(\mathbf{a}_k)\hat{\mathbf{x}}_k(t_i^+) \tag{2.48}$$

Note that each elemental controller gain is based on the corresponding assumed parameter value, $\mathbf{a}_k$, for that filter model. This form of Multiple Model Adaptive Control is a natural extension of the Bayesian Multiple Model Adaptive Estimator.

2.6.2 "Modified" MMAC Control

The "modification" in this method is an extension to the ad hoc technique of lower bounding the probabilities, $p_k(t_i)$, presented in Section 2.3.2. With this method, after each of the K control vectors are formed, the associated probabilities are compared to a set threshold, at a numerical value higher than the lower bound of that previous section. If any $p_k(t_i)$ is found to be below this threshold, the corresponding kth control vector, $\mathbf{u}_k(t_i)$, is discounted and not used to form the probabilistically weighted control vector. The probabilities are rescaled so their sum is one, and the remaining $\mathbf{u}_k(t_i)$ are used to form the control vector per Equation (2.47).

The use of the lower bounds described in Section 2.3.2 is still important to prevent the "lock-out" phenomenon associated with a particular $p_k(t_i)$ value being computed as essentially zero. However, if that $p_k(t_i)$ is artificially lower bounded, the control computed via Equation (2.47) will include a nonzero weighting on the corresponding inappropriate control, $\mathbf{u}_k(t_i)$. In this particular problem, it is readily possible that putting any nonzero weight on that $\mathbf{u}_k(t_i)$ could drastically degrade performance and could even cause closed-loop instability. Thus, the lower bounding method is incorporated to maintain viable $p_k(t_i)$ computations, but any $\mathbf{u}_k(t_i)$ associated with a $p_k(t_i)$ that has been artificially bounded is deemed to be unsuitable for use in control generation. Using a threshold somewhat higher than the lower bound value allows other $\mathbf{u}_k(t_i)$'s that are poor candidate controls (as indicated by their low $p_k(t_i)$ values) also to be eliminated from the control calculation.

2.6.3 *MMAC with MAP Control*

The method corresponds directly with the MAP method of state estimation (Section 2.3.2); the control is based solely on the elemental LQG controller with the highest associated probability.

2.6.4 *Single Fixed-Gain Control*

This method utilizes the fact that full-state feedback controllers are inherently robust and determines the control input based on a single nominal parameter value and a single state estimate as output from the MMAE [5:39-40, 22:93]. Figure 2-1 illustrates this concept, and the control law is given as:

$$\mathbf{u}^*(t_i) = -\overline{\mathbf{G}}_c^*(\mathbf{a}_{\text{nom}})\hat{\mathbf{x}}(t_i^+) \tag{2.49}$$

Hentz noted that determining a good value for $\mathbf{a}_{\text{nom}}$ may not be a trivial task [5:40].

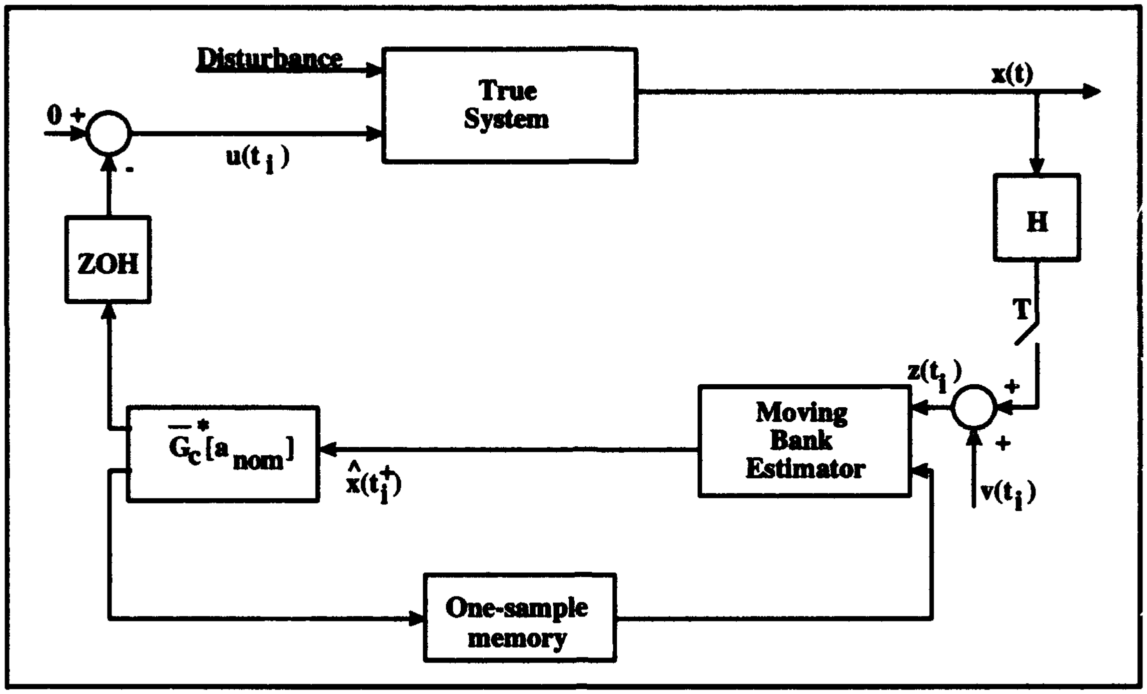

Figure 2-1. Single Fixed-Gain Controller

2.6.5 Single Changeable-Gain Control

This method is similar to the previous in that it, too, uses the single state estimate as output from the MMAE. However, the controller gain is computed based on the parameter estimate as output from the MMAE instead of a single nominal parameter value. This idea is illustrated in Figure 2-2, and the control law is given by:

$$\mathbf{u}^*(t_i) = -\overline{\mathbf{G}}_c^*[\hat{\mathbf{a}}(t_i)]\hat{\mathbf{x}}(t_i^+) \tag{2.50}$$

This requires that gain values be stored for each of the K parameters. Then, $\overline{\mathbf{G}}_c^*[\hat{\mathbf{a}}(t_i)]$ is computed by interpolation from the stored values[5:37-38].

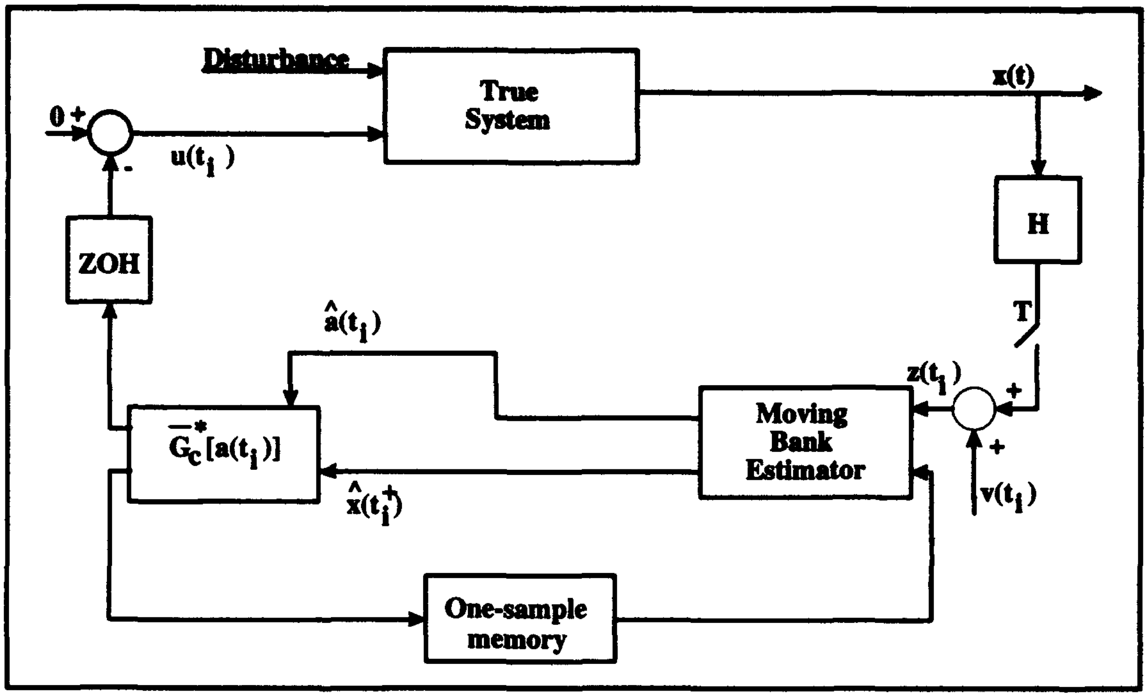

Figure 2-2. Single Changeable-Gain Controller

2.6.6 *"Modified" Single Changeable-Gain Control*

This method is similar to the previous method in that it also uses the parameter estimate from the MMAE, but the state estimate comes from a separate Kalman filter (not part of the moving bank of filters). This controller structure is illustrated in Figure 2-3, and the control law is defined the same as in the previous method with $\hat{\mathbf{x}}(t_i^+)$ from the MMAE replaced by $\hat{\mathbf{x}}(t_i^+)$ from the separate Kalman filter. This form of control can be useful if $\hat{\mathbf{a}}(t_i)$ is between two discrete parameter values for which elemental filters are designed. A single filter based on $\hat{\mathbf{a}}(t_i)$ might provide *substantially* better performance than *either* of those two elemental filters (or any linear combination of those filters); this form of algorithm allows a *coarser* discretization of the parameter space [18]. This method of control also reduces the possibility of control inputs based on an underestimated undamped natural frequency, which previous research has shown to cause instabilities in this type of system [5, 22:95, 30].

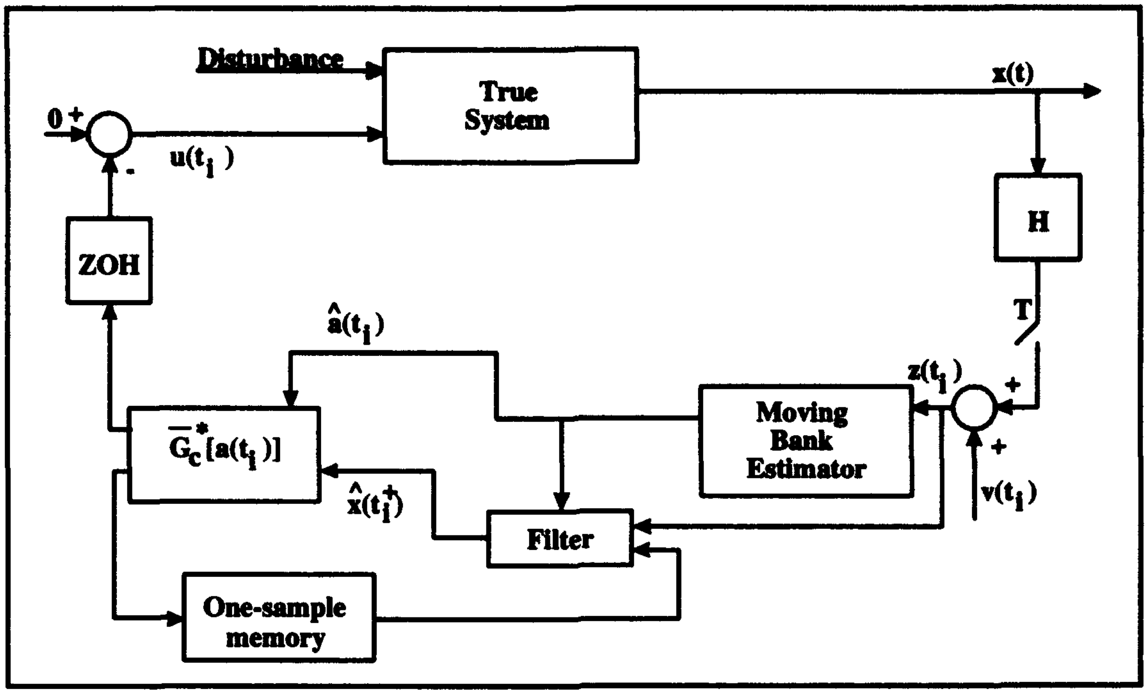

Figure 2-3. Modified Single Changeable-Gain Controller

2.7 Dithering Techniques

As indicated in Chapter 1, this research will apply dithering techniques in order to enhance parameter identification. Within the area of dithering, there are two basic methods: purposeful excitation designed for enhancement of parameter identification and purposeful commands to accomplish slewing of the space structure. Both types of commands are referred to as "open loop" control, since each are precomputable functions of time and not of states or measurements. If an appropriate excitation frequency is used, the system should be excited enough to enhance parameter identification. Likewise, giving the structure a slewing motion will cause a certain amount of natural system excitation to occur, which may also enhance parameter identification.

This section will discuss the different types of purposeful excitation dithering to be used in this research. The rigid-body states and how they are incorporated into the structural and control matrices, **F** and **B**, for use with purposeful (slew) commands, will be presented in Chapter 3.

2.7.1 Sinusoidal Dither

Three sinusoidal dither wave forms will be considered in this research. They are: (1) sine wave, (2) swept frequency sine wave, and (3) square wave. The sine wave would be a logical choice if the system's *identification-enhancing frequency* were known a priori. This optimum frequency is one which (when used to excite the system) should allow the best parameter identification to be made. If, however, this optimum frequency is not known, a method of using a sine wave while slowly changing the frequency across a particular range (swept frequency sine wave) could be employed [18]. This method would allow the system's identification-enhancing frequency to "show itself", and the dither could then be switched to a sine wave at that particular frequency (or be swept in some *small* frequency band about that frequency).

An alternative to using a pure sine wave dither is to use a square wave. With a square wave, not only is the fundamental frequency of the wave induced into the system, but additional harmonics (high frequency content) are also induced [18]. These additional harmonics may help parameter identification when the optimal dither frequency is not known.

When implementing sinusoidal type signals in a digital computer simulation, certain limitations exist and are discussed in the next section.

2.7.2 Dither Frequency

In a discrete-time system which is propagated forward at fixed intervals, $t_i - t_{i-1} = \Delta t$, Shannon's sampling theorem will apply when computing an input dither signal in the form of a $\sin(\omega t(t_i))$ function. In general, Shannon's sampling theorem requires that the sampling rate be at least twice the highest frequency of interest in the signal being sampled. This limit is also known as the Nyquist rate [27: 81]. For our application of computing a dither signal in a simulation which contains a particular *given* sampling rate, the limitation is placed on the frequency of the dither signal which can be produced. Thus, the maximum frequency in a sinusoid function that can be correctly produced in a fixed sampling rate simulation is:

$$f < \frac{1}{2\Delta t} \tag{2.51}$$

where f is highest frequency or Nyquist rate (in Hertz) with which the $\sin(\omega t(t_i))$ computation will produce a useful result. (Note that $\omega = 2\pi f$, if ω is in rad/sec.) Thus, computing $\sin(\omega t(t_i))$ with the limitation set in Equation (2.51) is analogous to sampling some continuous signal of frequency, f, with the Nyquist sampling rate requirement $f_s > 2f$, where f_s is the sampling rate. For the system used in this research $\Delta t = 0.01\,\text{sec}$, and therefore, $f < 50\,\text{Hz}$ or $\omega < 314.16\,\text{rad/sec}$. The importance of this maximum frequency will be seen in Chapter 4 when the actual dither frequencies are chosen.

One additional note concerns the actual application and computation of the sine wave and square wave dither inputs. Dither is applied via the control variable, $\mathbf{u}(t_i)$, by:

$$u(t_i) = A\sin(\omega t_i) \qquad \text{(sine wave)} \tag{2.52}$$

$$u(t_i) = A\,\mathrm{sgn}(\sin(\omega t_i)) \qquad \text{(square wave)} \tag{2.53}$$

where A is the amplitude of the wave, and the signum function "sgn" merely returns a +1 or a -1, depending on the polarity of the sine wave.

2.7.3 *Wide Band Noise Dither*

An alternative to using a sinusoid type of dither signal is to use wide band noise. With wide band noise, a shaping filter is designed for the purpose of providing a signal that covers a specified range of frequencies at a specified strength or power spectral density (PSD) value (over all inclusive frequencies). With such a shaping filter, the dither input to the system can provide a very rich frequency content, since a broad range of frequencies is used.

For this research a simple second-order shaping filter with two real poles will be used. Figure 2-4 shows the transfer function and equivalent block diagram design for this shaping filter. The PSD plot which represents the output of this transfer function and block diagram is shown as Figure 2-5. The corresponding state-space equations are as follows:

$$\begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \end{bmatrix} = \begin{bmatrix} -f_1 & 0 \\ 1 & -f_2 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} + \begin{bmatrix} w(t) \\ 0 \end{bmatrix} \tag{2.54}$$

$$n(t) = K\dot{x}_2(t) = K(x_1(t) - f_2 x_2(t)) \tag{2.55}$$

where w is zero mean, unity strength white noise, and K is an adjustable gain. f_1 and f_2 are the lower and upper frequency ranges of interest. Equation (2.54) can be written as the continuous-time matrix differential equation:

$$\dot{\mathbf{x}}(t) = \mathbf{F}\mathbf{x}(t) + \mathbf{w}(t) \tag{2.56}$$

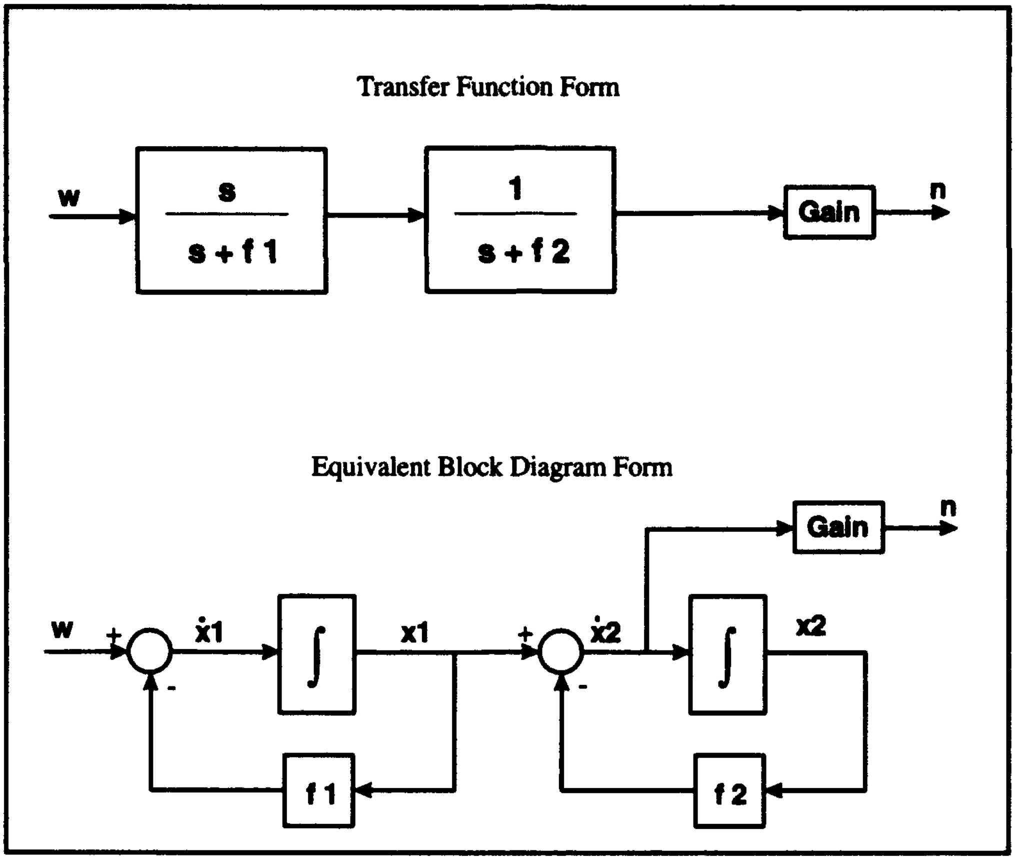

Figure 2-4. Block Diagram for Wide Band Noise

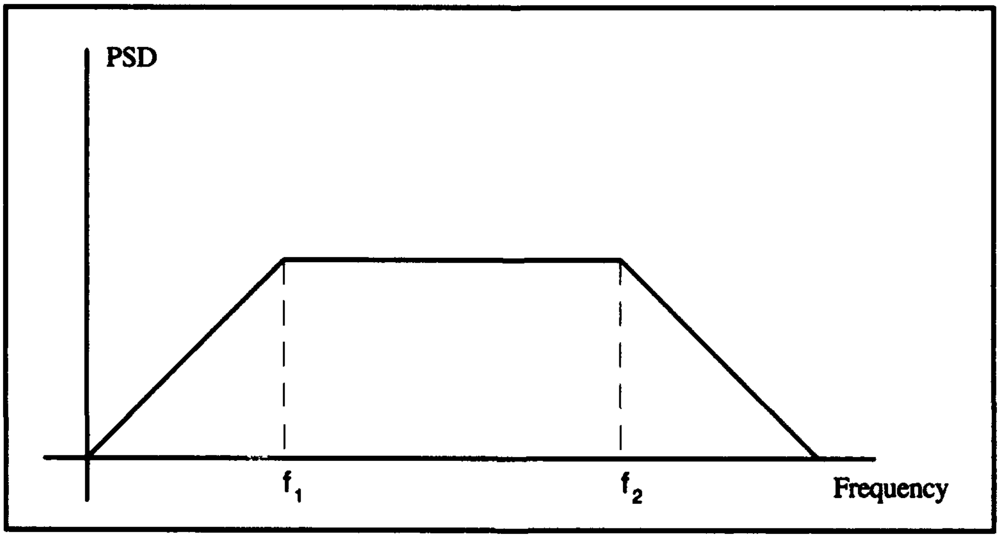

Figure 2-5. PSD Plot for Wide Band Noise

For discrete-time implementation, Equation (2.56) is written as:

$$\mathbf{x}(t_i) = \mathbf{\Phi}(t_i - t_{i-1})\mathbf{x}(t_{i-1}) + \mathbf{w}_d(t_{i-1}) \tag{2.57}$$

where $\mathbf{\Phi}(t_i - t_{i-1})$ is the state transition matrix associated with $\mathbf{F}$ and the statistics for $\mathbf{w}_d$ can be computed according to Equations (2.11), (2.12) and (2.13). For constant sampling period ($t_i - t_{i-1} = \text{const} = \Delta t$), the state transition matrix is simply:

$$\mathbf{\Phi}(\Delta t) = \begin{bmatrix} e^{-\Delta t f_1} & 0 \\ \frac{1}{f_2 - f_1}\left(e^{-\Delta t f_1} - e^{-\Delta t f_2}\right) & e^{-\Delta t f_2} \end{bmatrix} \tag{2.58}$$

The entire gain of the output of the shaping filter is controlled by K in Equation (2.55) and will be adjusted by trial and error.

2.8 Mathematical Modeling Methods

This section will present the two coordinate forms, physical and modal. The transformation from the physical form to the more desirable modal form will also be presented, along with a technique for producing a reduced-order model based on the modal coordinate form.

Many systems (including the one with this research) contain a high-dimension truth model. This is very useful in computer simulations in order to represent the “real” world accurately. However, many on-line applications cannot perform with such high state dimensions. Thus, a reduced-order model as the basis of filter and controller designs is necessary which will satisfy the on-line computer limitations and which will also yield satisfactory performance.

2.7.1 *Physical Coordinate Form*

The governing second-order differential equation describing the flexible structure vibrations is given by [7:39, 13:4, 32]:

$$\mathbf{M}\ddot{\mathbf{r}}(t)+\mathbf{C}\dot{\mathbf{r}}(t)+\mathbf{K}\mathbf{r}(t)=\mathbf{F}_1(\mathbf{u},t)+\mathbf{F}_2(t) \tag{2.59}$$

where:

- $\mathbf{r}(t)$ = n-dimensional vector representing the structure's physical position
- $\mathbf{M}$ = n-by-n constant mass matrix
- $\mathbf{C}$ = n-by-n constant damping matrix
- $\mathbf{K}$ = n-by-n constant stiffness matrix
- $\mathbf{F}_l(t)$ = r-dimensional deterministic control inputs
- $\mathbf{F}_2(t)$ = r-dimensional disturbances and unmodeled control inputs

With the assumptions that the deterministic control and noises enter the system linearly, and that the external disturbances, $\mathbf{F}_2(t)$, can be modeled as white Gaussian noises, then Equation (2.59) becomes [7:40, 31:4]:

$$\mathbf{M}\ddot{\mathbf{r}}(t)+\mathbf{C}\dot{\mathbf{r}}(t)+\mathbf{K}\mathbf{r}(t)=-\mathbf{b}\mathbf{u}(t)-\mathbf{g}\mathbf{w}(t) \tag{2.60}$$

where:

- $\mathbf{u}(t)$ = r-dimensional vector of actuator inputs
- $\mathbf{b}$ = n-by-r control input matrix identifying position of actuators and relationship between actuators and controlled variables
- $\mathbf{w}(t)$ = s-dimensional vector representing the dynamics driving noise

- $\mathbf{g}$ = n-by-s noise input matrix identifying position of input disturbances and relationship between the dynamics driving noise and the controlled variables

Equation (2.60) can be written as the stochastic differential equation in state-space form [7:40, 13:4]:

$$\dot{\mathbf{x}}(t) = \mathbf{F}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) + \mathbf{G}\mathbf{w}(t) \tag{2.61}$$

Note that this equation is of the same form as the system described by Equation (2.1) in Section 2.2 with the exception of constant matrices. The common state vector representation is given as:

$$\mathbf{x}(t) = \begin{bmatrix} \dot{\mathbf{r}}(t) \\ \mathbf{r}(t) \end{bmatrix}_{2n \times 1} \tag{2.62}$$

and the form of the constant system matrices is [2:3-20, 7:41]:

$$\mathbf{F} = \begin{bmatrix} -\mathbf{M}^{-1}\mathbf{C}_{n \times n} & -\mathbf{M}^{-1}\mathbf{K}_{n \times n} \\ \mathbf{I}_{n \times n} & \mathbf{0}_{n \times n} \end{bmatrix}_{2n \times 2n} \tag{2.63}$$

$$\mathbf{B} = \begin{bmatrix} -\mathbf{M}^{-1}\mathbf{b}_{n \times r} \\ \mathbf{0}_{n \times r} \end{bmatrix}_{2n \times r} \tag{2.64}$$

$$\mathbf{G} = \begin{bmatrix} -\mathbf{M}^{-1}\mathbf{g}_{n \times s} \\ \mathbf{0}_{n \times s} \end{bmatrix}_{2n \times s} \tag{2.65}$$

For the state vector described in Equation (2.62), the discrete-time measurement model of position and velocity is given by [2:3-20, 7:41]:

$$\mathbf{z}(t_i) = \left\{ \begin{bmatrix} \mathbf{H}_v & \mathbf{0} \\ \mathbf{0} & \mathbf{H}_p \end{bmatrix}_{m \times 2n} \mathbf{x}(t_i) \right\} + \mathbf{v}(t_i) \tag{2.66}$$

where:

- $m =$ number of measurements
- $\mathbf{v}(t_i) =$ m-dimensional measurement uncertainty modeled as a discrete-time white Gaussian noise of covariance $\mathbf{R}(t_i)$
- $\mathbf{H}_p$ = $(m/2)$-by-n position measurement matrix in physical coordinates
- $\mathbf{H}_v$ = $(m/2)$-by-n velocity measurement matrix in physical coordinates

Note that, for simplicity, an equal number of position and velocity measurements is assumed. The actual measurement matrix may vary, depending on available measurements, and does not affect the structure of the system matrices of Equations (2.63) through (2.65).

From this development it is seen that the physical coordinate form produces highly coupled system equations. With this coupled format, important system characteristics may not be easily identified, and thus, a more desirable form is needed. The next section will discuss the transformation from physical to modal coordinate form.

2.7.2 Modal Coordinate Form

Transforming the equations to modal form decouples the modes and makes the identification process simpler. For the purposes of this research, the damping matrix, $\mathbf{C}$, is assumed to be a linear combination of the mass and stiffness matrices [13:4]:

$$\mathbf{C} = \alpha\mathbf{M} + \beta\mathbf{K} \tag{2.67}$$

However, the calculation of α and β is not necessary when transforming to the modal coordinate form, as will be seen shortly. Given the new modal coordinate vector, $\tilde{\mathbf{r}}$, the transformation from physical form to modal form is described by [13:5]:

$$\mathbf{r} = \boldsymbol{T}\tilde{\mathbf{r}} \tag{2.68}$$

where $\boldsymbol{T}$ is a n-by-n transformation matrix determined from the system eigenvectors by solving [13:5, 32]:

$$\omega^2\mathbf{M}\boldsymbol{T} = \mathbf{K}\boldsymbol{T} \tag{2.69}$$

where the values for ω that satisfy this equation are referred to as the natural or modal frequencies.

Now, using the transformation Equation (2.68) to operate on the original system Equation (2.61), the resulting state space equation is given by [13:5]:

$$\dot{\tilde{\mathbf{x}}}(t) = \tilde{\mathbf{F}}\tilde{\mathbf{x}}(t) + \tilde{\mathbf{B}}\mathbf{u}(t) + \tilde{\mathbf{G}}\mathbf{w}(t) \tag{2.70}$$

where the transformed state vector from Equation (2.62) is now defined as [13:5]:

$$\tilde{\mathbf{x}}(t) = \begin{bmatrix} \dot{\tilde{\mathbf{r}}}(t) \\ \tilde{\mathbf{r}}(t) \end{bmatrix}_{2n\times 1} \tag{2.71}$$

and the transformed matrices from Equation (2.61) as applied to Equation (2.60) are defined as [2:3-22, 13:5]:

$$\tilde{\mathbf{F}} = \begin{bmatrix} -T^{-1}\mathbf{M}^{-1}\mathbf{C}T & -T^{-1}\mathbf{M}^{-1}\mathbf{K}T \\ \mathbf{I} & \mathbf{0} \end{bmatrix}_{2n \times 2n} \tag{2.72}$$

$$\tilde{\mathbf{B}} = \begin{bmatrix} -T^{-1}\mathbf{M}^{-1}\mathbf{b} \\ \mathbf{0} \end{bmatrix}_{2n \times r} \tag{2.73}$$

$$\tilde{\mathbf{G}} = \begin{bmatrix} -T^{-1}\mathbf{M}^{-1}\mathbf{g} \\ \mathbf{0} \end{bmatrix}_{2n \times s} \tag{2.74}$$

Since the modal form provides for independent equations, the resulting modal vectors, i.e., eigenvectors of $\tilde{\mathbf{F}}$, are orthogonal. This fact plus the following relationships [7:44, 32:1769]:

$$-T^{-1}\mathbf{M}^{-1}\mathbf{C}T = \left[-2\zeta_i\omega_i\right] \tag{2.75}$$

$$-T^{-1}\mathbf{M}^{-1}\mathbf{K}T = \left[-\omega_i^2\right] \tag{2.76}$$

allows the dynamics matrix to be written as follows:

$$\tilde{\mathbf{F}} = \begin{bmatrix} \left[-2\zeta_i\omega_i\right] & \left[-\omega_i^{2}\right] \\ \mathbf{I} & \mathbf{0} \end{bmatrix}_{2n \times 2n} \tag{2.77}$$

where the matrix is now block diagonal in terms of the undamped natural frequency and the damping ratio of the *i*-th mode. In Equations (2.75) through (2.77), the notation "$[a_i]$" denotes a diagonal matrix with a_1 through a_n running down the diagonal. Note that, because of the relationship in Equation (2.75), the damping matrix **C** does not need to be determined and thus the α and β in Equation (2.67) need not be known. The transformed measurement equation from Equation (2.66) may be written as [2:3-22]:

$$\tilde{\mathbf{z}}(t_i) = \left\{ \begin{bmatrix} \mathbf{H}_v \boldsymbol{T} & \mathbf{0} \\ \mathbf{0} & \mathbf{H}_p \boldsymbol{T} \end{bmatrix}_{m \times 2n} \tilde{\mathbf{x}}(t_i) \right\} + \mathbf{v}(t_i) \tag{2.78}$$

From this development it can be seen that the modal coordinate form does provide a decoupled set of equations from which to work. This decoupled form (modal form) allows direct insight into the physical system parameters (natural frequency and damping ratio), unlike with the physical coordinate form.

The next section presents the development of the *Modal Reduction Technique* as presented by Kokotovic [9] and Gustafson [2]. There are many other methods of providing a reduced-order model from a truth model, but Gustafson obtained the best results using this method [2:5-30], and so this technique will be pursued in this research.

2.7.3 Modal Reduction Technique

This state reduction technique will eliminate the higher frequency bending modes of the system. The concept behind forming a reduced-order model by eliminating these higher frequency modes (states) comes from the assumption that the higher frequency modes will become quiescent in a negligibly small amount of time. This assumption comes from the physical shape of the SPICE structure, which is two bulkheads connected by tripod legs. With this type of structure, it is suspected that low frequency mode effects will dominate.

The state-space system model from Equation (2.61) can be partitioned as follows [9:123]:

$$\dot{\mathbf{x}}(t)=\begin{bmatrix}\dot{\mathbf{x}}_1(t)\\ \dot{\mathbf{x}}_2(t)\end{bmatrix}=\begin{bmatrix}\mathbf{F}_{11} & \mathbf{F}_{12}\\ \mathbf{F}_{21} & \mathbf{F}_{22}\end{bmatrix}\begin{bmatrix}\mathbf{x}_1(t)\\ \mathbf{x}_2(t)\end{bmatrix}+\begin{bmatrix}\mathbf{B}_1\\ \mathbf{B}_2\end{bmatrix}\mathbf{u}(t)+\begin{bmatrix}\mathbf{G}_1\\ \mathbf{G}_2\end{bmatrix}\mathbf{w}(t) \qquad (2.79)$$

where the system is driven by deterministic controls, $\mathbf{u}(t)$, and zero-mean, white Gaussian noise $\mathbf{w}(t)$ of strength $\mathbf{Q}(t)$. The upper partition, $\mathbf{x}_1(t)$, corresponds to the low frequency modes to be maintained, and the lower partition, $\mathbf{x}_2(t)$, corresponds to the high frequency modes to be removed.

Assuming instantaneous steady state in the higher modes ($\dot{\mathbf{x}}_2(t)=\mathbf{0}$), these modes can be eliminated with negligible impact to overall system performance. The lower partition differential equation is set equal to zero by [9:123]:

$$\dot{\mathbf{x}}_2(t)=\mathbf{F}_{21}\mathbf{x}_1(t)+\mathbf{F}_{22}\mathbf{x}_2(t)+\mathbf{B}_2\mathbf{u}(t)+\mathbf{G}_2\mathbf{w}(t)=\mathbf{0} \qquad (2.80)$$

Since $\mathbf{F}_{21}$ and $\mathbf{F}_{22}$ are square matrices and assuming $\mathbf{F}_{22}^{-1}$ exists, $\mathbf{x}_2(t)$ can now be written in terms of $\mathbf{x}_1(t)$ and system inputs:

$$\mathbf{x}_2(t)=-\mathbf{F}_{22}^{-1}\left[\mathbf{F}_{21}\mathbf{x}_1(t)+\mathbf{B}_2\mathbf{u}(t)+\mathbf{G}_2\mathbf{w}(t)\right] \qquad (2.81)$$

Substituting Equations (2.80) and (2.81) into Equation (2.79) results in [9:123]:

$$\dot{\mathbf{x}}_1(t)=\left[\mathbf{F}_{11}-\mathbf{F}_{12}\mathbf{F}_{22}^{-1}\mathbf{F}_{21}\right]\mathbf{x}_1(t)+\left[\mathbf{B}_1-\mathbf{F}_{12}\mathbf{F}_{22}^{-1}\mathbf{B}_2\right]\mathbf{u}(t)+\left[\mathbf{G}_1-\mathbf{F}_{12}\mathbf{F}_{22}^{-1}\mathbf{G}_2\right]\mathbf{w}(t) \qquad (2.82)$$

Analogous to other developments in this Chapter, the equivalent discrete-time model will be developed. The equivalent discrete-time model of Equation (2.79) is given as [2:3-24]:

$$\mathbf{x}(t_{i+1}) = \begin{bmatrix} \mathbf{x}_1(t_{i+1}) \\ \mathbf{x}_2(t_{i+1}) \end{bmatrix} = \begin{bmatrix} \mathbf{\Phi}_{11} & \mathbf{\Phi}_{12} \\ \mathbf{\Phi}_{21} & \mathbf{\Phi}_{22} \end{bmatrix} \begin{bmatrix} \mathbf{x}_1(t_i) \\ \mathbf{x}_2(t_i) \end{bmatrix} + \begin{bmatrix} \mathbf{B}_{\mathrm{d}1} \\ \mathbf{B}_{\mathrm{d}2} \end{bmatrix} \mathbf{u}(t_i) + \begin{bmatrix} \mathbf{G}_{\mathrm{d}1} \\ \mathbf{G}_{\mathrm{d}2} \end{bmatrix} \mathbf{w}_{\mathrm{d}}(t_i) \tag{2.83}$$

Similar to the steady-state assumption for the continuous-time case $(\dot{\mathbf{x}}_2(t) = \mathbf{0})$, the discrete-time steady-state assumption becomes $(\mathbf{x}_2(t_{i+1}) = \mathbf{x}_2(t_i))$. Thus, the higher-order modes of Equation (2.83) are represented as [2:3-24]:

$$\mathbf{\Phi}_{21}\mathbf{x}_1(t_i) + [\mathbf{\Phi}_{22} - \mathbf{I}]\mathbf{x}_2(t_i) + \mathbf{B}_{\mathrm{d}2}\mathbf{u}(t_i) + \mathbf{G}_{\mathrm{d}2}\mathbf{w}_{\mathrm{d}}(t_i) = \mathbf{0} \tag{2.84}$$

$$\mathbf{x}_2(t_i) = -[\mathbf{\Phi}_{22} - \mathbf{I}]^{-1}[\mathbf{\Phi}_{21}\mathbf{x}_1(t_i) + \mathbf{B}_{\mathrm{d}2}\mathbf{u}(t_i) + \mathbf{G}_{\mathrm{d}2}\mathbf{w}_{\mathrm{d}}(t_i)] \tag{2.85}$$

In order to evaluate these discrete-time matrices from the continuous-time matrices, a first-order approximation method is used as follows [2:3-24, 19:172, 357]:

$$\mathbf{\Phi}_{22} = \mathbf{I} + \mathbf{F}_{22}\Delta t \tag{2.86}$$

$$\mathbf{\Phi}_{21} = \mathbf{F}_{21}\Delta t \tag{2.87}$$

$$\mathbf{B}_{\mathrm{d}2} = \mathbf{B}_2\Delta t \tag{2.88}$$

$$\mathbf{G}_{\mathrm{d}2} = \mathbf{G}_2\Delta t \tag{2.89}$$

$$\mathbf{Q}_{\mathrm{d}} = \mathbf{Q} / \Delta t \tag{2.90}$$

This last equation comes from the first-order approximation of $\mathbf{G}_{\mathrm{d}2}\mathbf{Q}_{\mathrm{d}}\mathbf{G}_{\mathrm{d}2}^{\mathrm{T}} \cong \mathbf{G}_2\mathbf{Q}_2\mathbf{G}_2^{\mathrm{T}}\Delta t$ and $\mathbf{G}_{\mathrm{d}2} = \mathbf{G}_2\Delta t$ [2:3-25, 18].

Substituting Equations (2.86) through (2.90) into Equation (2.85) results in:

$$\mathbf{x}_2(t_i) = -\left[\mathbf{F}_{22}\Delta t\right]^{-1}\left[\mathbf{F}_{21}\Delta t\mathbf{x}_1(t_i) + \mathbf{B}_2\Delta t\mathbf{u}(t_i) + \mathbf{G}_2\Delta t\mathbf{w}_d(t_i)\right] \tag{2.91}$$

where $\mathbf{w}_d(t_i)$ has covariance $\mathbf{Q}_d$. Eliminating Δt from this expression results in:

$$\mathbf{x}_2(t_i) = -\mathbf{F}_{22}^{-1}\left[\mathbf{F}_{21}\mathbf{x}_1(t_i) + \mathbf{B}_2\mathbf{u}(t_i) + \mathbf{G}_2\mathbf{w}_d(t_i)\right] \tag{2.92}$$

This is the same result obtained in the continuous-time case, Equation (2.81), with the exception of $\mathbf{w}_d(t_i)$ replacing $\mathbf{w}(t)$. Then, substituting Equation (2.92) into the discrete-time measurement equation:

$$\mathbf{z}(t_i) = \begin{bmatrix}\mathbf{H}_1 & \mathbf{H}_2\end{bmatrix}\begin{bmatrix}\mathbf{x}_1(t_i) \\ \mathbf{x}_2(t_i)\end{bmatrix} + \mathbf{v}(t_i) \tag{2.93}$$

and expanding, yields [2:3-25]:

$$\mathbf{z}(t_i) = \left[\mathbf{H}_1 - \mathbf{H}_2\mathbf{F}_{22}^{-1}\mathbf{F}_{21}\right]\mathbf{x}_1(t_i) - \mathbf{H}_2\mathbf{F}_{22}^{-1}\left[\mathbf{B}_2\mathbf{u}(t_i) + \mathbf{G}_2\mathbf{w}_d(t_i)\right] + \mathbf{v}(t_i) \tag{2.94}$$

The second term in this measurement equation, $-\mathbf{H}_2\mathbf{F}_{22}^{-1}\left[\mathbf{B}_2\mathbf{u}(t_i) + \mathbf{G}_2\mathbf{w}_d(t_i)\right]$, is a direct feedthrough term created by the order reduction [9:123].

One item yet to be determined in this modal reduction method is the break between the low and high frequency modes. This is accomplished by examining an ordered list of the modal frequencies and deciding what constitutes a natural break or jump in modal frequency. The actual break-point made comes from physical insight and desired computation load.

With the break-point known, the system dynamics matrix from Equation (2.77) is separated into an augmented form as [9:123]:

$$\tilde{\mathbf{F}} = \left[\begin{array}{cc|cc} \left[-2\zeta_1\omega_1\right] & \left[-\omega_1^2\right] & \mathbf{0} & \mathbf{0} \\ \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{0} & \left[-2\zeta_2\omega_2\right] & \left[-\omega_2^2\right] \\ \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} \end{array}\right] \tag{2.95}$$

The low frequency modes to be maintained are represented by the upper left quadrant, and the higher frequency modes to be removed are represented by the lower right quadrant. These two quadrants correspond to the $\mathbf{F}_{11}$ and $\mathbf{F}_{22}$ partitions in Equation (2.79) with the off-diagonal blocks, $\mathbf{F}_{12}$ and $\mathbf{F}_{21}$, set equal to zero. Substituting this information into Equations (2.82) and (2.94) yields [9:123]:

$$\begin{aligned} \dot{\tilde{\mathbf{x}}}_1(t) &= \tilde{\mathbf{F}}_{11}\tilde{\mathbf{x}}_1(t) + \tilde{\mathbf{B}}_1\mathbf{u}(t) + \tilde{\mathbf{G}}_1\mathbf{w}_r(t) \\ &= \tilde{\mathbf{F}}_r\tilde{\mathbf{x}}_1(t) + \tilde{\mathbf{B}}_r\mathbf{u}(t) + \tilde{\mathbf{G}}_r\mathbf{w}_r(t) \end{aligned} \tag{2.96}$$

$$\begin{aligned} \mathbf{z}(t_i) &= \tilde{\mathbf{H}}_1\tilde{\mathbf{x}}_1(t_i) - \tilde{\mathbf{H}}_2\tilde{\mathbf{F}}_{22}^{-1}\left[\tilde{\mathbf{B}}_2\mathbf{u}(t_i) + \tilde{\mathbf{G}}_2\mathbf{w}_d(t_i)\right] + \mathbf{v}_r(t_i) \\ &= \tilde{\mathbf{H}}_r\tilde{\mathbf{x}}_1(t_i) - \tilde{\mathbf{D}}_u\mathbf{u}(t_i) + \tilde{\mathbf{D}}_w\mathbf{w}_d(t_i) + \mathbf{v}_r(t_i) \end{aligned} \tag{2.97}$$

where the effect of $\mathbf{F}_{12} = \mathbf{0}$ and $\mathbf{F}_{21} = \mathbf{0}$ is seen explicitly, and where the subscript, r, denotes "reduced-order." Thus, the direct feedthrough matrices, $\tilde{\mathbf{D}}_u$ and $\tilde{\mathbf{D}}_w$, allow the effects of control and dynamics driving noise inputs to be "seen" in the reduced-order measurement.

2.9 Summary

In this chapter the fundamental concepts concerning the Kalman filter, the state and parameter estimator (the MMAE), and the stochastic controller (the MMAC) were presented. For the MMAE and MMAC, various algorithms and enhancements were presented that have been studied in the past. Shown below is a summary of the assumptions and particular estimator/controller options (particular algorithms) which are to be used in this research effort.

- Kalman filter: Based on linear, time-invariant system model driven by stationary noise.

- LQG controller: Constant state and control weighting matrices, no cross-weighting matrix.

- MMAE: ME/A and ME/I will be considered.
  Move logic will be Parameter Position Estimate Monitoring

- MMAC: "Modified" MMAC will be used.

Additionally, presented in this chapter were the dithering techniques to be used for enhancing parameter identification and the mathematical modeling for developing reduced-order filter models. The dither inputs are to consist of (1) square wave, (2) sine wave, (3) swept sine wave, and (4) wide band noise. The mathematical modeling consists of (1) the Physical Coordinate Form, (2) the Modal Coordinate Form, and (3) the Modal Reduction Technique. Chapter 3 will present the actual system structural model development to be used in this research.

3. System Development

3.1 Introduction

This chapter presents a summary of the SPICE 2 system model originally developed by Gustafson [2: Chapter 3]. The physical structure of the SPICE 2 model is described, followed by a description of the mathematical models used to simulate the actual structure and its disturbance inputs. The truth and filter models used in this research are also discussed.

The addition of rigid-body states is presented here as well. These are used to simulate planar motion of the structure so as to induce naturally occurring structural vibrations. This is an addition to Gustafson's development and is used to complement the research of applying dithering techniques for enhancing uncertain parameter identification.

3.2 SPICE Structure

The structure model used in this research is model 2 of the SPace Integrated Controls Experiment (SPICE) [14: Chapter 3]. The actual structure resides at Phillips Laboratory, Kirtland AFB, New Mexico and has subsequently been revised up to model 5. For the purposes of this research, however, the SPICE 2 model will suffice, since this research effort is devoted to applying and proving concepts within the area of adaptive filters and controllers.

3.2.1 Physical Description

The SPICE structure is composed of three major sections, as illustrated in Figure 3-1. The hexagonal base or bulkhead is 6.19 meters in diameter. This base supports the entire structure and also contains the primary mirror assembly mounted on its top. Three tripod legs connect the bulkhead to the top portion, which houses the Secondary Mirror Assembly (SMA), which is 1.32 meters in diameter. The overall height of the structure is 8.44 meters. The X-Y axes are centered and in the plane of the structure's base, with the Y-axis pointing towards tripod leg number 1. The Z-axis points straight up the middle of the structure and is the Line of Sight (LOS) axis. The 12 Proof Mass Actuators (PMAs) located on the tripod legs contain their own local coordinate frames, while the 6 PMAs located at the base are aligned along the Z-axis.

Keeping the secondary mirror assembly aligned with the primary mirror assembly is the main concern of the SPICE research. An exaggerated non-alignment is shown in Figure 3-2. For this research, pure torsional bending (about the Z-axis) is not considered, since it has only minor impact on the line-of-sight tracking error.

3.2.2 Actuators and Sensors

The actuators are the Proof Mass Actuator (PMA) type and are used to quell any vibration of the tripod legs or the bulkhead. The sensors used are accelerometers and are collocated with the PMAs. The accelerometers are used to measure the relative bending motion of the structure. Twelve PMAs and accelerometers are located on the tripod legs, four on each leg. On each tripod leg, two PMAs and accelerometers are located at both the one-third and two-third length positions. Six PMAs and accelerometers are located at the base of the structure. Thus, the structure contains 18 PMAs and 18 accelerometers.

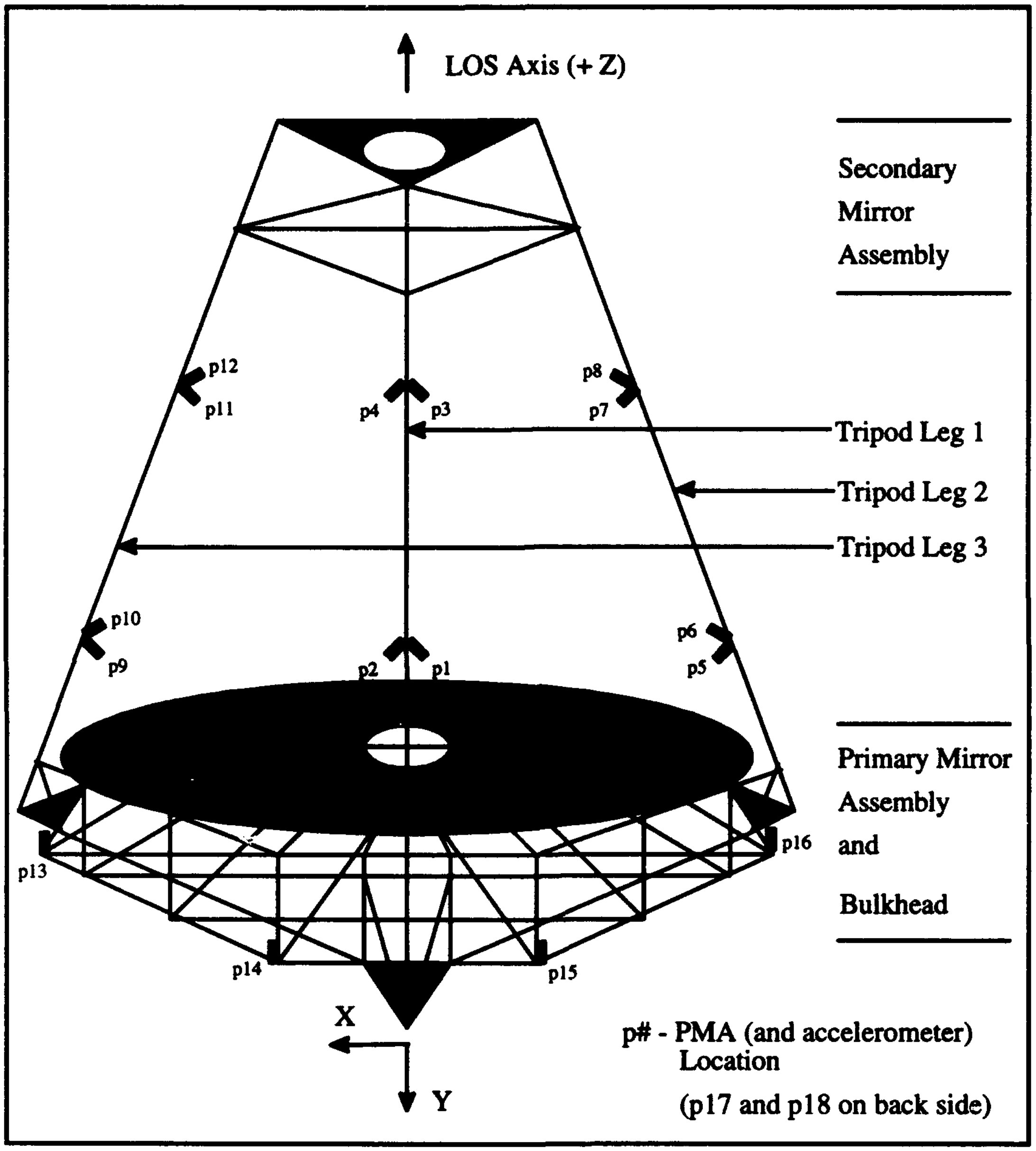
LOS Axis (+ Z)
Secondary Mirror Assembly
p12
p11
p4
p3
p8
p7
Tripod Leg 1
Tripod Leg 2
Tripod Leg 3
p10
p9
p2
p1
p6
p5
Primary Mirror Assembly and Bulkhead
p13
p16
p14
p15
X
Y
p# - PMA (and accelerometer) Location
(p17 and p18 on back side)

Figure 3-1. SPICE Structure [14: III-14]

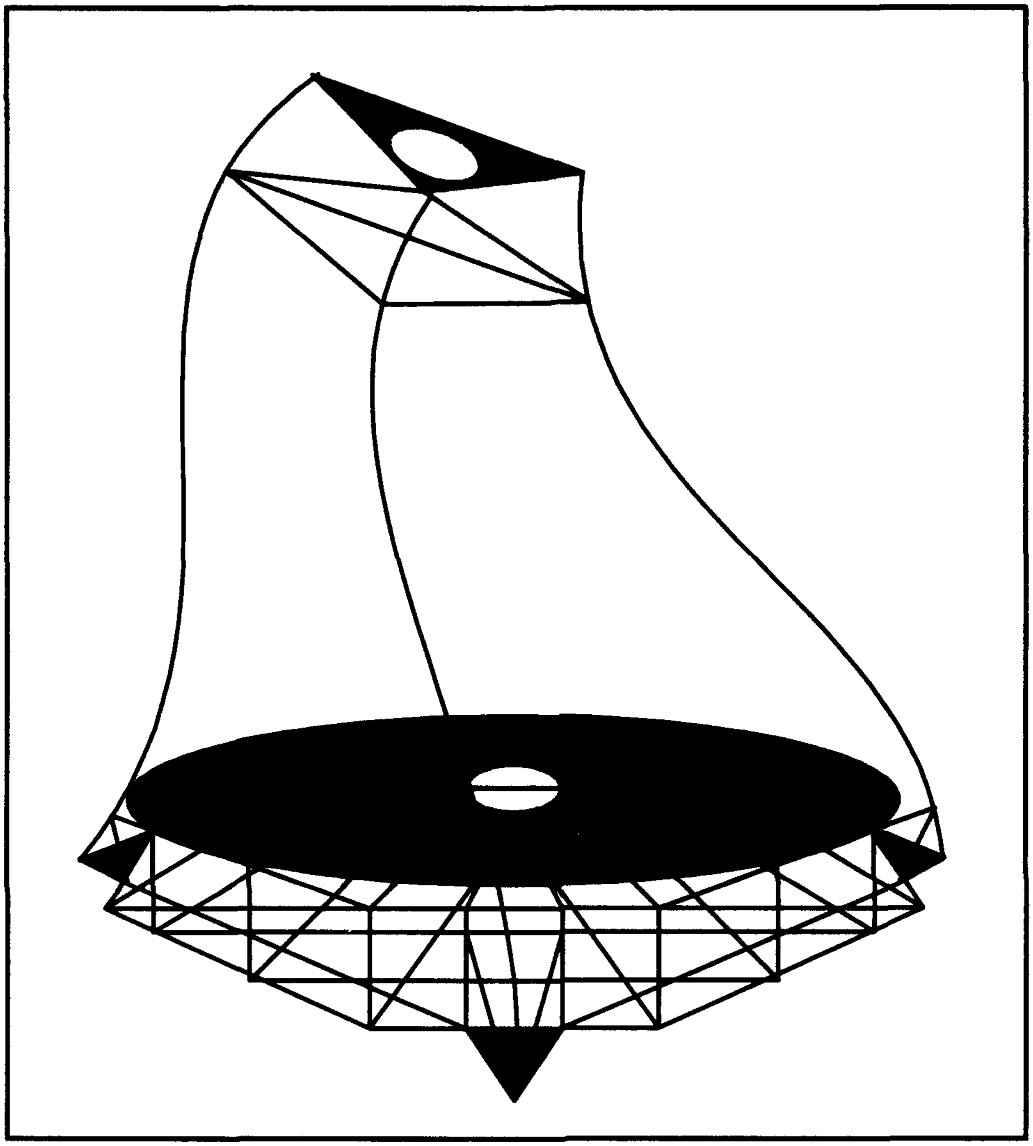

Figure 3-2. Flexible SPICE Structure

3.3 *System Mathematical Model Description*

The mathematical model of each component of the SPICE 2 system is presented here. The overall block diagram of the system model is shown in Figure 3-3. The Disturbance, PMA and Accelerometer blocks all contain shaping filters and state-space models for their respective functions. The LOS block is merely a transformation matrix which outputs the LOS coordinates, "x" and "y", which indicate the alignment/misalignment of the secondary mirror assembly with the primary mirror assembly. The Structure block contains the structural bending dynamics. As shown in Figure 3-3, the Structure block does not contain the augmented rigid-body dynamics. The addition of rigid-body states will be shown later in the chapter. The MMAC block represents the entire moving-bank multiple model algorithm for providing active control of the SPICE structure.

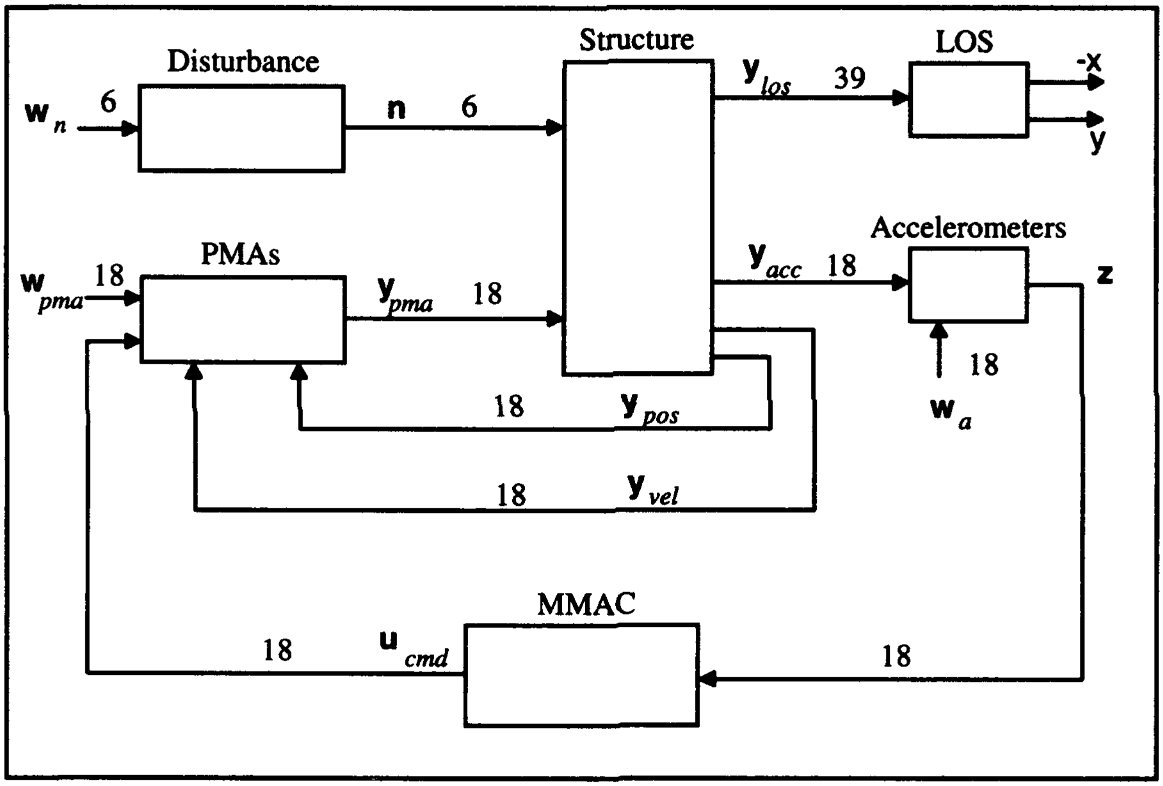

Figure 3-3. SPICE System Block Diagram [2: 3-5]

3.3.1 Disturbances

The structural disturbances are modeled as entering the structure at six points: at the top and bottom of each tripod leg. Details of how these disturbances impact the structure are given shortly in Equation (3.3). These are modeled as shaped noise over the

31.416 to 62.832 radians per second (*rps*) frequency range. This is modeled as 6 white noise inputs of equivalent strength, each shaped by a fourth-order notch filter, as shown in Figure 3-4. Thus, the disturbance model contributes 24 states to the system model. The state equation for this model is given as:

$$\dot{\mathbf{x}}_n(t) = \mathbf{F}_n \mathbf{x}_n(t) + \mathbf{G}_n \mathbf{w}_n(t) \tag{3.1}$$

where:

- $\mathbf{x}_n(t)$ = 24-state vector representing the disturbance states
- $\mathbf{F}_n$ = 24-by-24 constant plant matrix
- $\mathbf{G}_n$ = 24-by-6 constant noise input matrix
- $\mathbf{w}_n(t)$ = 6-by-1 unit-strength white Gaussian noise vector

The corresponding output equation is:

$$\mathbf{n}(t) = \mathbf{C}_n \mathbf{x}_n(t) \tag{3.2}$$

where:

- $\mathbf{n}(t)$ = 6-by-1 output colored noise vector
- $\mathbf{C}_n$ = 6-by-24 constant matrix

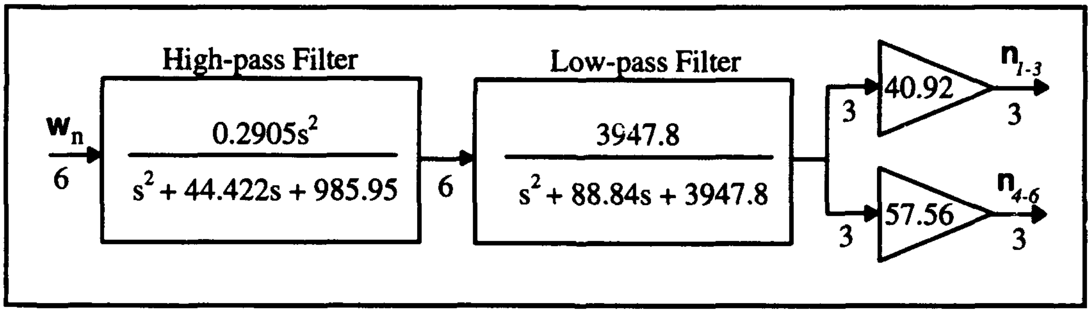
High-pass Filter
Low-pass Filter
$\mathbf{w}_n$
6
$\frac{0.2905s^2}{s^2 + 44.422s + 985.95}$
6
$\frac{3947.8}{s^2 + 88.84s + 3947.8}$
3
40.92
$\mathbf{n}_{1\text{-}3}$
3
3
57.56
$\mathbf{n}_{4\text{-}6}$
3

Figure 3-4. Disturbance Model Block Diagram [2: 3-6]

The output vector is defined as:

$$\mathbf{n} = \begin{bmatrix} n_1 \\ n_2 \\ n_3 \\ n_4 \\ n_5 \\ n_6 \end{bmatrix} = \begin{bmatrix} \text{SMA x disturbance Leg 1} \\ \text{SMA y disturbance Leg 2} \\ \text{SMA z disturbance Leg 3} \\ \text{Bulkhead z disturbance Leg 1} \\ \text{Bulkhead z disturbance Leg 2} \\ \text{Bulkhead z disturbance Leg 3} \end{bmatrix} \tag{3.3}$$

3.3.2 Proof Mass Actuators (PMAs)

Each of the 18 PMAs are modeled identically as shown in Figure 3-5. They are modeled as second order high-pass filters with a cutoff frequency of approximately 30 *rps*. Since the PMA design is second order, the 18 PMAs contribute 36 states to the system model. The state equation for the PMA model is given as:

$$\dot{\mathbf{x}}_{pma}(t) = \mathbf{F}_{pma}\mathbf{x}_{pma}(t) + \mathbf{B}_{pma}\mathbf{u}_{pma}(t) + \mathbf{G}_{pma}\mathbf{w}_{pma}(t) \tag{3.4}$$

where:

- $\mathbf{x}_{pma}(t) =$ 36-state vector representing the PMA filter states
- $\mathbf{F}_{pma} =$ 36-by-36 constant plant matrix
- $\mathbf{B}_{pma} =$ 36-by-54 constant deterministic input matrix
- $\mathbf{G}_{pma} =$ 36-by-18 constant noise input matrix
- $\mathbf{u}_{pma} =$ 54-by-1 "control" vector composed of components $\mathbf{u}_{cmd}$, $\mathbf{u}_{vel}$ and $\mathbf{u}_{pos}$
- $\mathbf{w}_{pma}(t) =$ 18-by-1 unit-strength white Gaussian noise vector

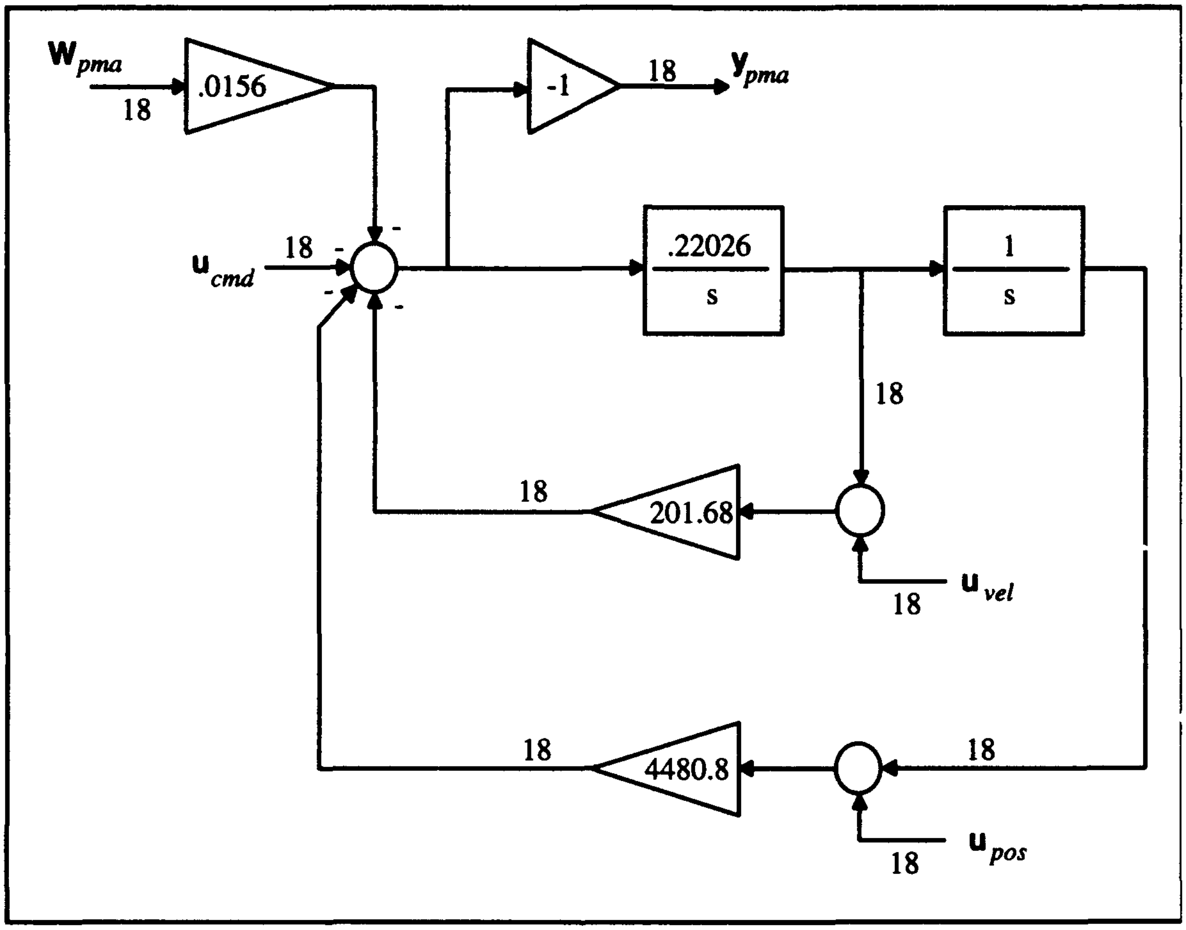

Figure 3-5. PMA Block Diagram

Note that $\mathbf{u}_{cmd}$ is the control output of the MMAC. Additionally, $\mathbf{u}_{vel}$ and $\mathbf{u}_{pos}$ are $\mathbf{y}_{vel}$ and $\mathbf{y}_{pos}$, respectively, of Figure 3-3, i.e., outputs of the "structure" block. These are the relative velocities and positions of each PMA mass, taken with respect to the point on the structure where the PMA is attached [2: Chapter 3 pg. 12].

The corresponding output equation is given by:

$$\mathbf{y}_{pma}(t) = \mathbf{C}_{pma}\mathbf{x}_{pma}(t) + \mathbf{D}_{\mathrm{u}\,pma}\mathbf{u}_{pma}(t) + \mathbf{D}_{\mathrm{w}\,pma}\mathbf{w}_{pma}(t) \tag{3.5}$$

where:

- $\mathbf{y}_{pma}(t)$ = 18-by-1 output response vector
- $\mathbf{C}_{pma}$ = 18-by-36 constant output matrix
- $\mathbf{D}_{\mathrm{u}\,pma}$ = 18-by-54 constant deterministic input direct feedthrough matrix
- $\mathbf{D}_{\mathrm{w}\,pma}$ = 18-by-18 noise direct feedthrough matrix

The direct feedthrough matrices result from an equal order over equal order filter transfer function. The control and noise input vectors, $\mathbf{u}_{pma}$ and $\mathbf{w}_{pma}$, respectively, are defined as:

$$\mathbf{u}_{pma} = \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_{18} \\ u_{19} \\ u_{20} \\ \vdots \\ u_{36} \\ u_{37} \\ u_{38} \\ \vdots \\ u_{54} \end{bmatrix} = \begin{bmatrix} \text{PMA 1 command} \\ \text{PMA 2 command} \\ \vdots \\ \text{PMA 18 command} \\ \text{PMA 1 velocity} \\ \text{PMA 2 velocity} \\ \vdots \\ \text{PMA 18 velocity} \\ \text{PMA 1 position} \\ \text{PMA 1 position} \\ \vdots \\ \text{PMA 18 position} \end{bmatrix} \tag{3.6}$$

and

$$\mathbf{w}_{pma} = \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_{18} \end{bmatrix} = \begin{bmatrix} \text{PMA 1 noise} \\ \text{PMA 2 noise} \\ \vdots \\ \text{PMA 18 noise} \end{bmatrix} \tag{3.7}$$

The output vector, $\mathbf{y}_{pma}$, is defined by:

$$\mathbf{y}_{pma} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_{18} \end{bmatrix} = \begin{bmatrix} \text{PMA 1 response} \\ \text{PMA 2 response} \\ \vdots \\ \text{PMA 18 response} \end{bmatrix} \tag{3.8}$$

3.3.3 Structure

This section first presents the equations associated with the flexible structure. This is followed by the augmentation of the flexible dynamics with the rigid-body dynamics. The desired coupling is formed from this augmentation in order to excite flexible bending motion of the structure when a rigid-body control is applied.

3.3.3.1 Flexible Dynamics

In general, the flexible dynamics of the structure are represented by the following state equation:

$$\dot{\mathbf{x}}_s(t) = \mathbf{F}_s\mathbf{x}_s(t) + \mathbf{B}_s\mathbf{y}_{pma}(t) + \mathbf{G}_s\mathbf{n}(t) \tag{3.9}$$

where

- $\mathbf{x}_s(t) =$ $2n$-by-1 state vector representing the structural flexible bending modes
- $n =$ number of structural flexible bending modes
- $\mathbf{F}_s$ = $2n$-by-$2n$ matrix
- $\mathbf{B}_s$ = $2n$-by-18 constant matrix

- $\mathbf{G}_s$ = 2n-by-6 constant matrix
- $\mathbf{n}$ is defined by Equations (3.2) and (3.3)
- $\mathbf{y}_{pma}$ is defined by Equations (3.5) and (3.8)

The associated output equation is defined as:

$$\mathbf{y}_s(t) = \mathbf{C}_s\mathbf{x}_s(t) + \mathbf{D}_{\mathrm{n}s}\mathbf{n}(t) + \mathbf{D}_{\mathrm{y}s}\mathbf{y}_{pma}(t) \tag{3.10}$$

where

- $\mathbf{y}_s = (t)$ 93-by-1 PMA output response vector
- $\mathbf{C}_s$ = 93-by-2n constant output matrix
- $\mathbf{D}_{\mathrm{n}s}$ = 93-by-6 noise direct feedthrough matrix
- $\mathbf{D}_{\mathrm{y}s}$ = 93-by-18 constant deterministic input direct feedthrough matrix

The corresponding output vector is defined as:

$$\mathbf{y}_s = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_{39} \\ y_{40} \\ y_{41} \\ \vdots \\ y_{57} \\ y_{58} \\ y_{59} \\ \vdots \\ y_{74} \\ y_{75} \\ y_{76} \\ \vdots \\ y_{93} \end{bmatrix} = \begin{bmatrix} \text{LOS optical element } 1,\ y_{los1} \\ \text{LOS optical element } 2,\ y_{los2} \\ \vdots \\ \text{LOS optical element } 39,\ y_{los39} \\ \text{PMA } 1 \text{ acceleration},\ y_{acc1} \\ \text{PMA } 2 \text{ acceleration},\ y_{acc2} \\ \vdots \\ \text{PMA } 18 \text{ acceleration},\ y_{acc18} \\ \text{PMA } 1 \text{ velocity},\ y_{vel1} \\ \text{PMA } 2 \text{ velocity},\ y_{vel2} \\ \vdots \\ \text{PMA } 18 \text{ velocity},\ y_{vel18} \\ \text{PMA } 1 \text{ position},\ y_{pos1} \\ \text{PMA } 2 \text{ position},\ y_{pos2} \\ \vdots \\ \text{PMA } 18 \text{ position},\ y_{pos18} \end{bmatrix} \tag{3.11}$$

The first 39 elements comprise the $\mathbf{y}_{los}$ vector, as indicated in Figure 3-3. These LOS outputs are obtained from an optical scoring system which uses 39 separate laser and sensor pairs to detect changes in position of the secondary mirror assembly with respect to the primary mirror assembly. These LOS outputs are multiplied by a LOS transformation matrix in order to obtain the "x" and "y" line-of-sight errors, as indicated in Figure 3-3. The $\mathbf{y}_{acc}$ output vector (elements 40 through 57) indicates the relative acceleration of each PMA mass with respect to the point on the structure where the PMA is attached. These acceleration outputs are used by the accelerometer model, yet to be discussed in Section 3.3.4. The remaining velocity and position outputs are similar to the acceleration outputs and were discussed in Section 3.3.2.

As indicated by Equation (3.9), the flexible structure modeling adds $2n$ states to the system model. The next section will address the augmentation of rigid-body states.

3.3.3.2 Rigid-body Dynamics

The rigid-body motion considered in this research is planar motion. Thus, two additional states are required. In polar coordinate terms, these are angular position and angular rate. For the purposes of this research with the SPICE 2 flexible structure, this motion is confined to be perpendicular to the Z axis, rotating about an axis in the X-Y plane. The axis of rotation is located at the base (bulkhead) of the structure. Refer to Figure 3-6 for an illustration of a typical rigid-body rotation.

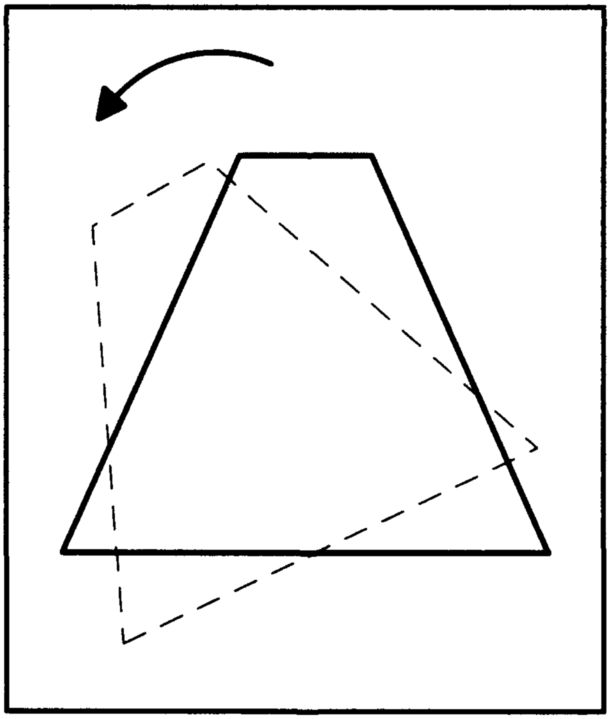

Figure 3-6. Rigid-Body Rotational Motion

The two-dimensional rigid-body dynamics can be described by:

$$M(t) = I\ddot{\theta}(t) \tag{3.12}$$

where $M(t)$ is the moment causing the rotation to occur, I is the constant moment of inertia matrix about the point of rotation, and $\ddot{\theta}(t)$ is the angular acceleration. Put into state space form, this becomes:

$$\begin{bmatrix} \dot{x}_{1rb}(t) \\ \dot{x}_{2rb}(t) \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} x_{1rb}(t) \\ x_{2rb}(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 1/I \end{bmatrix} u_{rb}(t) \tag{3.13}$$

where

- $x_{1rb}(t)$ = rigid-body angle position = $\theta(t)$
- $x_{2rb}(t)$ = rigid-body angular rate = $\dot{\theta}(t)$
- $u_{rb}(t)$ = rigid-body control applied = $M(t)$

In more general terms, Equation (3.13) can be written as:

$$\dot{\mathbf{x}}_{rb}(t) = \mathbf{F}_{rb}\mathbf{x}_{rb}(t) + \mathbf{b}_{rb}u_{rb}(t) \tag{3.14}$$

When the rigid-body dynamics are augmented to the flexible dynamics, Equation (3.9) becomes:

$$\begin{bmatrix} \dot{\mathbf{x}}_s(t) \\ \dot{\mathbf{x}}_{rb}(t) \end{bmatrix} = \begin{bmatrix} \mathbf{F}_s & \mathbf{0} \\ \mathbf{0} & \mathbf{F}_{rb} \end{bmatrix} \begin{bmatrix} \mathbf{x}_s(t) \\ \mathbf{x}_{rb}(t) \end{bmatrix} + \begin{bmatrix} \mathbf{B}_s & \mathbf{b}_{pma} \\ \mathbf{0} & \mathbf{b}_{rb} \end{bmatrix} \begin{bmatrix} \mathbf{y}_{pma}(t) \\ u_{rb}(t) \end{bmatrix} + \begin{bmatrix} \mathbf{G}_s \\ \mathbf{0} \end{bmatrix} \mathbf{n}(t) \tag{3.15}$$

Note that an additional column is augmented to the flexible structure $\mathbf{B}_s$. The $\mathbf{b}_{pma}$ vector is simply one particular column of the $\mathbf{B}_s$ matrix. This comes from the way in which the rigid-body control is applied to the structure. For ease of simulation, rigid-body control is applied at the structure's base at one of the PMA locations. (PMA location number 13 was used in this research; recall Figure 3-1.) In order for the *flexible* dynamics of the system to be realized from the action of applying a force to a particular location on the structure, the appropriate column of $\mathbf{B}_s$ must be included in the augmentation, so that it is correctly multiplied by $u_{rb}(t)$.

3.3.4 Accelerometers

Each of the 18 accelerometers is modeled as a first order high-pass filter with an additive corruptive shaped white noise. The noise filter is modeled as a second order high-pass filter. The block diagram for the accelerometer design is shown in Figure 3-7.

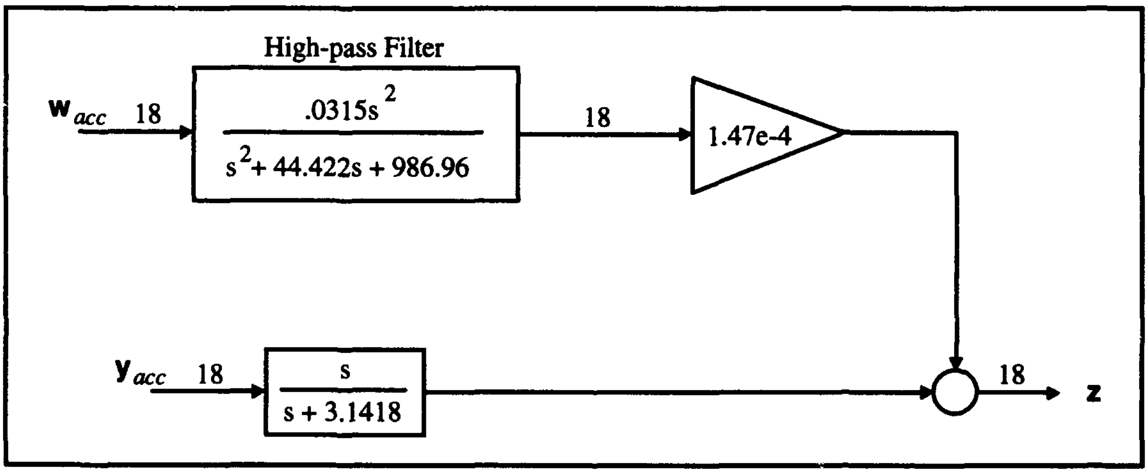

Figure 3-7. Accelerometer Design

With 18 inputs, the accelerometer model contributes 54 states to the system model. The state equation for the accelerometer design is given as:

$$\dot{\mathbf{x}}_{acc}(t)=\begin{bmatrix}\dot{\mathbf{x}}_a(t)\\ \dot{\mathbf{x}}_{\mathrm{n}a}(t)\end{bmatrix}=\begin{bmatrix}\mathbf{F}_a & \mathbf{0}\\ \mathbf{0} & \mathbf{F}_{\mathrm{n}a}\end{bmatrix}\begin{bmatrix}\mathbf{x}_a(t)\\ \mathbf{x}_{\mathrm{n}a}(t)\end{bmatrix}+\begin{bmatrix}\mathbf{B}_a\\ \mathbf{0}\end{bmatrix}\mathbf{y}_{acc}(t)+\begin{bmatrix}\mathbf{0}\\ \mathbf{G}_{\mathrm{w}a}\end{bmatrix}\mathbf{w}_{acc}(t) \qquad (3.16)$$

where

- $\mathbf{x}_a(t)$ = 18-state vector representing the accelerometer response states
- $\mathbf{x}_{\mathrm{n}a}(t)$ = 36-state vector representing time-correlated accelerometer noise
- $\mathbf{F}_a$ = 18-by-18 constant accelerometer plant matrix
- $\mathbf{F}_{\mathrm{n}a}$ = 36-by-36 constant accelerometer noise plant matrix
- $\mathbf{B}_a$ = 18-by-18 constant input matrix
- $\mathbf{G}_{\mathrm{w}a}$ = 36-by-18 constant noise input matrix
- $\mathbf{y}_{acc}(t)$ is defined by elements 40 through 57 of Equation (3.11)
- $\mathbf{w}_{acc}$ = 18-by-1 unit-strength white Gaussian noise vector

The output of the accelerometers is the measurements and is given by the measurement equation:

$$\mathbf{z}(t_i)=\begin{bmatrix}\mathbf{H}_a & \mathbf{H}_{\mathrm{n}a}\end{bmatrix}\begin{bmatrix}\mathbf{x}_a(t_i)\\ \mathbf{x}_{\mathrm{n}a}(t_i)\end{bmatrix}+\mathbf{D}_{yacc}\mathbf{y}_{acc}(t_i)+\mathbf{D}_{\mathrm{w}a}\mathbf{w}_{acc}(t_i) \qquad (3.17)$$

where

- $\mathbf{x}_a(t_i)$ = 18-state vector representing the accelerometer response states
- $\mathbf{x}_{\mathrm{n}a}(t_i)$ = 36-state vector representing time-correlated accelerometer noise
- $\mathbf{H}_a$ = 18-by-18 constant accelerometer measurement matrix

- $\mathbf{H}_{na}$ = 18-by-36 constant accelerometer noise measurement matrix
- $\mathbf{D}_{ya}$ = 18-by-18 constant direct feedthrough matrix
- $\mathbf{D}_{wa}$ = 36-by-18 constant direct feedthrough matrix
- $\mathbf{y}_{acc}(t)$ is defined by elements 40 through 57 of Equation (3.11)
- $\mathbf{w}_{acc}$ = 18-by-1 unit-strength white Gaussian noise vector

The direct feedthrough matrices in the second and third terms of Equation (3.17) are a result of the accelerometer model containing transfer functions with equal order numerator and denominator.

3.4 Truth Model and Filter Model

The system model presented thus far contains $114 + 2n$ states. The 114 states are comprised of PMA, accelerometer, and input disturbance and accelerometer measurement noise states. The structure contributes the $2n$ states for n flexible modes. The truth model used by Gustafson [2] will be employed here, and that is with a 40-mode structure model. This gives 194 states for the truth model. Using 40 flexible bending modes as the truth model creates a model with undamped natural frequencies below 100 *Hz*, which is the frequency limit of the research model. A 194-state truth model is also computationally feasible for simulation purposes.

The filter model consists of the same truth model states with the shaping noise states all replaced by white noise. This reduces the number of states to $54 + 2n = 134$ with 40 flexible bending modes. Gustafson also investigated other reduced-order filter model designs based on the techniques of modal reduction and internally balanced model reduction. The MMAC did not perform well with these reduced-order filter designs, however. This poor controller performance may have been partially due to the low number of bending modes utilized (6 bending modes). Current research by Schiller [28] with the subsequent SPICE 4 model has utilized 26 bending modes in a reduced-order filter and controller, which has produced excellent MMAC performance. While not all of the improvement in controller performance may be attributed to the higher number of bending modes utilized, this does certainly warrant another look with the SPICE 2 model. Thus, another reduced-order filter model consisting of 26 bending modes (52 bending mode states) + 54 PMA and accelerometer states (106 total states) will be considered in this research. The reduced-order filter model will be constructed using the modal reduction method described in Chapter 2. Particularly once more precise measurements are assumed, this reduced-order design model may well yield good performance.

3.5 Summary

This chapter has presented a brief discussion of the SPICE 2 system model used for this research. This included a description of the physical structure, along with a look at each subsystem. The mathematical models for input disturbances, PMAs, the flexible and rigid-body dynamics and accelerometer measurements were presented. The truth model was shown to consist of 194 states while the filter models were shown to consist of 134 and 106 states.

The next chapter will discuss all aspects of the various software programs used for simulating the MMAE and MMAC algorithms.

4. Simulation

4.1 Introduction

This chapter presents a discussion of the software and the computer simulation plan used to analyze the effectiveness of the moving-bank MMAE and MMAC algorithms as applied to the SPICE 2 structure model. A Monte Carlo analysis is the main instrument used to show this effectiveness and is performed using the MMAE/MMAC software developed from previous thesis research [1, 2, 5, 7, 11, 26, 29, 32]. The formation of the discretized parameter space and the changes made to the existing software will also be discussed in this chapter.

4.2 Monte Carlo Analysis

A Monte Carlo analysis is performed in order to evaluate the performance of the moving-bank MMAE and MMAC algorithms. This is accomplished by using the simulation software (see Section 4.3) to produce multiple-run sample statistics of the data of interest (estimation errors for filter performance and line-of-sight deviations for controller performance, as well as parameter estimation errors for both cases). The main emphasis of the simulations will be with the 194-state truth model versus the 134-state filter model, although one comparison is made with a 106-state reduced-order filter model (see Chapter 3).

The simulations performed can be broken into two areas: estimator analysis and controller analysis. These two are depicted in Figure 4-1 with variables defined as:

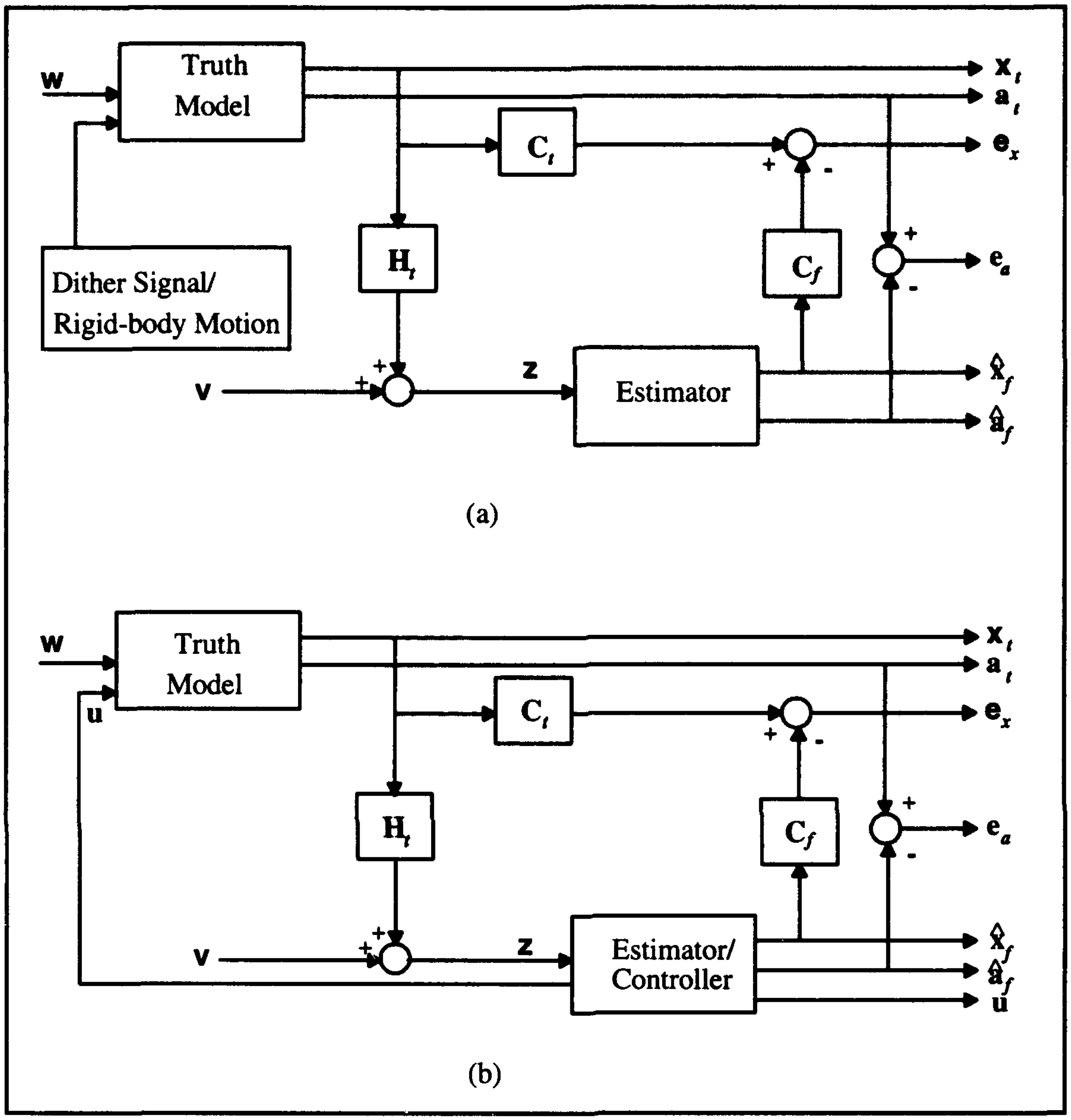

Figure 4-1. (a) Estimator Simulation, and (b) Controller Simulation [29]

- $\mathbf{x}_t(t_i)$ = truth model states
- $\hat{\mathbf{x}}_f(t_i)$ = filter estimates of the system states
- $\mathbf{a}_t(t_i)$ = parameter vector implemented in the truth model
- $\hat{\mathbf{a}}_f(t_i)$ = filter estimates of the uncertain parameter vector
- $\mathbf{e}_a(t_i)$ = error in the parameter estimate defined as: $\mathbf{e}_a(t_i) = \mathbf{a}_t(t_i) - \hat{\mathbf{a}}_f(t_i)$
- $\mathbf{e}_x(t_i)$ = error in the system estimate

The following sections present the error vector formulation and the error vector statistics.

4.2.1 Error Vector Formulations

The X and Y Line-of-Sight (LOS) deviations are the concern in this research. Therefore, for the estimator performance evaluation, an error vector is formed by subtracting the filter-determined X and Y LOS values from the corresponding truth-model-calculated values. This is shown in Figure 4-1(a) and by the following equation:

$$\mathbf{e}_x(t_i) = \mathbf{C}_t \mathbf{x}_t(t_i) - \sum_{j=1}^{K} \mathbf{C}_{f_j} \hat{\mathbf{x}}_{f_j}(t_i) \cdot p_j(t_i) \tag{4.1}$$

where $\mathbf{C}_t$ and $\mathbf{C}_{f_j}$ are the output matrices used to determine the X and Y LOS deviations for the truth model and j^{th} filter model respectively. The summation in this equation is computed by use of the conditional probabilities from each of the filters in the moving bank (as presented in Chapter 2). The actual form of this error vector is given by:

$$\mathbf{e}_x(t_i) = \begin{bmatrix} e_{x_1}(t_i) \\ e_{x_2}(t_i) \end{bmatrix} = \begin{bmatrix} \text{X} - \text{axis LOS position error} \\ \text{Y} - \text{axis LOS position error} \end{bmatrix} \tag{4.2}$$

For the controller performance evaluation (closed loop performance), the item of primary concern is how well the LOS deviations are quelled to zero. Thus, the vector of importance is given by the following equation:

$$\mathbf{e}'_x(t_i) = \mathbf{C}_t \mathbf{x}_t(t_i) \tag{4.3}$$

where $\mathbf{e}'_x(t_i)$ is the vector of the true X and Y LOS deviations (position errors). The control vector $\mathbf{u}(t_i)$ in Figure 4-1b also warrants investigation, to ensure that commanded controls are of reasonable magnitude. In both estimation and control, $\mathbf{e}_a$ of Figure 4-1 also merits inspection, to determine how effectively the algorithm estimates the uncertain parameter $\mathbf{a}$ in its attempt to provide adaptive estimation or control of states.

4.2.1 Error Vector Statistics

The mean or average value of the error vectors just defined is one of two critical statistics, and the main statistic used for portraying parameter estimation performance, in this research. The mean is given by: [19: 129]:

$$E\{\mathbf{e}_x(t_i)\} \approx \mathbf{M}_{e_x}(t_i) = \frac{1}{L}\sum_{l=1}^{L} \mathbf{e}_{x_l}(t_i) \tag{4.4}$$

where L is the number of Monte Carlo runs made, and $\mathbf{e}_{x_l}(t_i)$ is the value of the error vector during the l^{th} simulation at run time t_i. The covariance of the error signal is calculated by [19: 130]:

$$\mathbf{P}_{e_x}(t_i) = E\left\{\left[\mathbf{e}_x(t_i) - E\{\mathbf{e}_x(t_i)\}\right]\left[\mathbf{e}_x(t_i) - E\{\mathbf{e}_x(t_i)\}\right]^{\mathrm{T}}\right\}$$

$$\approx \frac{1}{L-1}\sum_{l=1}^{L}\left\{\mathbf{e}_{x_l}(t_i)\mathbf{e}_{x_l}^{\mathrm{T}}(t_i)\right\} - \frac{L}{L-1}\mathbf{M}_{e_x}(t_i)\mathbf{M}_{e_x}^{\mathrm{T}}(t_i) \tag{4.5}$$

Equations (4.4) and (4.5) represent general formulas for calculating error vector statistics and are used to calculate the mean and covariance of $\mathbf{e}'_x(t_i)$ and $\mathbf{e}_a(t_i)$ as well.

Additionally, this research will use $L = 10$ Monte Carlo runs to generate the sample statistics.

The error vector statistics described above will be useful for creating plots of the desired variables. However, in order to quantify the performance of the controller in a compact manner, suitable for tabulating, another statistic is used. This is the temporal average of the RMS value and is calculated by [2: Chapter 4 pg. 4]:

$$M_{tp} = \frac{1}{N} \sum_{j=i-N+1}^{i} \sqrt{\left[\mathbf{M}_{\mathbf{e}'_x}(t_j)\mathbf{M}^{\mathrm{T}}_{\mathbf{e}'_x}(t_j) + \mathbf{P}_{\mathbf{e}'_x}(t_i)\right]_{pp}} \quad \text{for } p = 1 \text{ and } 2 \tag{4.6}$$

where M_{tp} is the temporal average of the p^{th} component of $\mathbf{e}'_x$ and N is the number of sample periods over which the temporal averaging is accomplished. For this research, the temporal average will be taken over the last three seconds of the run duration.

4.3 Simulation Software

In the most recent study by Gustafson [2], the software (composed of MATRIXx [16] and FORTRAN programs) which was "handed down" from previous research [1, 5, 7, 11, 26, 29, 32] was modified for the SPICE 2 model. The software is still resident on the Sun workstations and is split into three sections: (1) preprocessor, (2) processor, and (3) postprocessor. Each of these is discussed below as to its function and the changes made during this research.

4.3.1 Preprocessor

Inherited Preprocessor. The preprocessor is actually two separate processors: a MATRIXx portion and a FORTRAN portion. MATRIXx is used to generate the discretized truth and filter model matrices as well as the steady-state filter and controller gain matrices. These matrices are then stored in files (one file for each truth and filter model based on a particular discretized parameter value) to be read in by the preprocessor FORTRAN program, which merely "collects" all the information for each parameter model and stores the variables into two files (one for the truth models and one for the filter models) to be read by the processor.

Current Preprocessor. During this research it was found that the MATRIXx routine for obtaining the steady-state controller matrix, $\overline{\mathbf{G}}_c^*$, did not converge to the proper value. Investigation into this discrepancy showed that the MATRIXx routine did converge properly for the simple example in the MATRIXx manual. Further investigation showed that the MATLAB [15] routine for solving the steady-state gain matrix did converge properly for the SPICE 2 model. Thus, all the preprocessor MATRIXx coding was converted to MATLAB. As will be shown in Chapter 5, even though the MATRIXx routine failed to converge to the proper controller gain, identical results for the closed-loop temporal average values were obtained as compared to Gustafson's [2]. Additionally, the residual covariance calculation and its inverse, $\mathbf{A}_k$ and $\mathbf{A}_k^{-1}$ of Equation (2.29), were removed from the FORTRAN portion of the code and put into MATLAB. Problems persisted in the past with this matrix inverse calculation, but the MATLAB routine showed no problems. One other major change to the preprocessor was to convert the FORTRAN code to double precision arithmetic to allow for greater numerical accuracy. The double precision routines were compiled with the IMSL library [6].

4.3.2 Processor

Inherited Processor: This set of FORTRAN subroutines performs the Monte Carlo simulation. This is done by propagating the truth and filter models, with dynamics and measurement noises simulated by appropriate scaling of the outputs of a unity variance random number generator available in FORTRAN. The different types of bank movement and MMAE/MMAC options are performed as well. The various variables to be analyzed are output to separate files for each sample time and run. These files are then read in by the postprocessor.

Current Processor: A few changes were made to the processor for enhancement and to allow for the analysis required in this research. First, the conversion to double precision was made, as with the preprocessor. Second, code was added for applying open-loop dither inputs. Options were made available via an input data file for selecting square wave, sine wave (fixed and swept frequency), or wide band noise. Third, the program was modified to allow for rigid-body states to be included, augmented to the bending mode states originally simulated. This was accomplished for the purpose of simulating rigid-body slew commands in order to analyze the effects on uncertain parameter identification enhancement.

Two other changes were made to the processor due to inadequacies discovered. The first concerns the bounds placed on the probability calculation. From the discussion in Chapter 2, there must be a lower bound placed on the calculated p_k probabilities. The value for this bound was set to 0.1, which was too high. The lower bound was reset to 0.01. Additionally, it was found that the code allowed for an *upper* bound to be placed on the calculated probability as well. This upper bound was removed all together. The second change concerns the probability density calculation. It was found that, if the residuals were too large, the exponential term in Equation (2.27) was driven to zero, due to finite word length with the digital computer. In many situations, this would occur for each of the k filters, resulting in the entire denominator of Equation (2.26) being evaluated as zero, preventing the probabilities from being calculable. So, a check was put in the code, such that, if the density calculation produced zero, it was reset to a very small value (to approximate a near-zero value and allow the probability calculation to continue).

4.3.3 Post Processor

Inherited Post Processor: This FORTRAN program calculates all the pertinent statistics used for analyzing the MMAE and MMAC performance. These statistics are output to separate files to be used by a compatible plotting routine.

Current Post Processor: The only change to the post processor was the conversion to double precision. MATLAB was used to plot the data output by the post processor.

4.4 Analysis Plan

This research effort is directed towards improving the parameter identification performance by use of various dither input signals prior to application of closed-loop control. The secondary issues are threefold: (1) possible satisfactory performance with a 26-bending mode reduced-order filter model, (2) possible controller performance improvement via better accelerometer measurements (less measurement noise), and (3) possible estimator/controller/parameter identification performance improvement via an alternate discretized parameter space (different from that used by Gustafson [2]).

Before any of these issues are explored, however, the software and system models will be verified by attempting to reproduce the single filter benchmark results obtained by Gustafson [2]. Additionally, throughout the dither signal analysis, a comparison will be made between the ME/A and ME/I methods of probability density calculation (discussed in Chapter 2). The next three sections discuss in further detail the issues concerning input dither frequency, improved accelerometer measurements, and parameter space discretization.

4.4.1 Dither Frequency Selection

Based on the research accomplished by Hanlon [4], the flexible structure modal frequencies are the basis for selecting the dither frequencies. The modal frequencies correspond to the system eigenvalues and are useful, since inputs based on these frequencies should excite the system more than by using some other arbitrary frequency. The modal frequencies for the first 30 modes are listed in Table 4-1. In order to determine which frequency or frequencies are best suited for exciting the system, the dither input frequency will be swept through the range of modal frequencies using a sinusoid signal. It is expected that a dither frequency corresponding to one of the lower bending modes will excite the system most effectively, due to the physical characteristics of the space structure – it being two rigid bulkhead assemblies attached by tripod legs. Once this "best" dither frequency is determined, it will be used as the basis for square wave and wide band dither inputs as well.

4.4.2 Improved Accelerometer Measurements

In this analysis, an attempt will be made to provide a benchmark with a "perfect" measurement model. This will be accomplished by changing the gain in Figure 3-7 to zero. This will have the affect of removing all accelerometer measurement noise. Next, the original gain will be reduced by one and two orders of magnitude in order to simulate a possible realistic improvement in the measurement device. In each of these cases it will be necessary to modify (retune) the original $\mathbf{R}_f$ matrix values for optimal results.

From Table 4-1, note that the first 26 modal frequencies are less than 50 Hz. Recall from Section 2.7.2 that, with a sampling time of 0.01 sec, the maximum useful sinusoid dither frequency is 50 Hz. Thus, the dither input frequencies used in this research will be limited to the first 26 modal frequencies. This may not be an actual limitation, however, since it is expected that the optimum dither frequency will reside within the lower modal frequencies.

Mode	**Eigenvalue**	**Damping Ratio**	**Natural Frequency (Hz)**
1	-0.09729 ± 48.64j	0.002	7.7418
2	-0.1115 ± 55.77j	0.002	8.8762
3	-0.1115 ± 55.77j	0.002	8.8763
4	-0.2318 ± 115.9j	0.002	18.446
5	-0.2318 ± 115.9j	0.002	18.449
6	-0.2350 ± 117.5j	0.002	18.720
7	-0.3357 ± 167.8j	0.002	26.712
8	-0.3604 ± 180.2j	0.002	28.681
9	-0.3605 ± 180.2j	0.002	28.684
10	-0.3793 ± 189.6j	0.002	30.180
11	-0.4103 ± 205.2j	0.002	32.654
12	-0.4106 ± 205.3j	0.002	32.673
13	-0.4164 ± 208.2j	0.002	33.135
14	-0.4166 ± 208.3j	0.002	33.150
15	-0.4168 ± 208.4j	0.002	33.168
16	-0.4624 ± 231.2j	0.002	36.798
17	-0.4626 ± 231.3j	0.002	36.816
18	-0.4689 ± 234.4j	0.002	37.312
19	-0.5407 ± 270.3j	0.002	43.025
20	-0.5409 ± 270.4j	0.002	43.040
21	-0.5628 ± 281.4j	0.002	44.784
22	-0.5697 ± 284.8j	0.002	45.334
23	-0.5698 ± 284.9j	0.002	45.343
24	-0.6116 ± 305.8j	0.002	48.669
25	-0.6215 ± 310.7j	0.002	49.455
26	-0.6216 ± 310.8j	0.002	49.467
27	-0.6724 ± 336.2j	0.002	53.505
28	-0.6894 ± 344.7j	0.002	54.864
29	-0.7004 ± 350.2j	0.002	55.732
30	-0.7008 ± 350.4j	0.002	55.770

Table 4-1. Modal Eigenvalues and Natural Frequencies for the First 30 Modes

4.4.3 Parameter Space Discretization (Sensitivity Analysis)

A sensitivity analysis is performed in order to produce a set of discrete parameter points. This is accomplished by setting the truth and filter models to the nominal parameter value, i.e., the nominal values for the undamped natural frequencies of the bending modes. Simulations are then run in this single filter mode with closed loop control applied. On each successive run, a scalar multiplier on all the undamped natural frequencies in the truth model is varied by a small amount, δ, until the closed loop control becomes unstable. The change in each undamped natural frequency is given by:

$$\omega_{new} = (1+\delta)\omega_{old} \tag{4.7}$$

Once the value for δ is found which produces unstable control, a new parameter point is found by reducing this value for δ by 10 %. Then, the truth and filter models are reset with the new values for the undamped natural frequencies, and the process is repeated in order to obtain a new value for δ and a new parameter point. This entire process is repeated until the desired range in undamped natural frequency is obtained. Parameter points are also generated by changing the modal frequencies in the opposite direction by:

$$\omega_{new} = (1-\delta)\omega_{old} \tag{4.8}$$

This method of performing a sensitivity analysis to obtain the discrete parameter points allows for the widest spread between each point, without any two adjacent points being so far apart so as to produce unstable closed-loop control. Also, this would yield residuals in filters at adjacent parameter points that would have sufficiently different characteristics that the p_k computation of Equations (2.26) to (2.29) would not be incapacitated by such residuals being virtually indistinguishable from one another.

However, for reasons discussed in Chapter 5, this method was used only to obtain the first point, displaced upward in frequency from the nominal values. The resulting δ was 0.0025. Twenty-nine parameter points were then generated in approximately the same span as used by Gustafson [2: Chapter 5, pg. 35], each with a scalar multiplier spacing of 0.0025.

4.5 Summary

This chapter has presented a discussion of the simulation analysis to be performed in this research. The Monte Carlo simulation has been described, along with the appropriate error vector formulations. Also included in this discussion has been the analysis plan which has shown the main points of the research to be accomplished, along with a detailed look at the formation of the discretized parameter space.

The next chapter will discuss the results of this research.

5. Results

5.1 Introduction

Research was conducted in accordance with the analysis plan presented in Chapter 4. This research was based on the theory and procedures presented in the first three chapters as well. This chapter presents the results of this research. As discussed in Chapters 1 and 4, the areas to be investigated are:

- Use dithering techniques to enhance uncertain parameter identifiability, prior to engaging closed-loop control
- Investigate the effects of purposeful rigid-body slew commands in regard to enhancing uncertain parameter identifiability
- Investigate an alternative parameter space discretization
- Investigate the effects of increased measurement sensor precision
- Investigate the effectiveness of a 106-state modal reduced-order filter/controller

Throughout this presentation of results, reference will be made to graphs located in Appendix A. These figures include three types: (1) is the mean ± one standard deviation ($\pm\sigma$) of the error between the truth model values and filter estimates. This is computed by Equations (4.4) and (4.5), utilizing the error vector in Equation (4.1), and is referred to as *estimation error*. (2) is the mean ± one standard deviation ($\pm\sigma$) of the actual LOS error. This is computed by Equations (4.4) and (4.5), utilizing the error vector in Equation (4.3), and is referred to as *LOS error*. (3) is the mean of the filter-computed parameter location. The true parameter variation is shown coincidentally so as to indicate the effectiveness of the multiple model adaptive algorithm in estimating the undamped natural frequency of the bending modes.

5.2 *MMAE/MMAC "Best Case" Performance*

This section presents the results from a single and a multiple (three) filter simulation with an unchanging parameter value. The single filter represents the artificially informed filter (non-adaptive case). The multiple filter unchanging parameter value case is presented in comparison, to demonstrate the ability of the MMAE/MMAC algorithm to maintain "lock" on a good estimate of parameters, and thus to maintain state estimation and/or control performance that is comparable to the artificially informed benchmark.

5.2.1 *Single Filter Baseline*

A duplication of Gustafson's work was accomplished with the 194-state truth model vs. the 134-state filter model. The purpose of this simulation with a single filter was two-fold. First, it provides a verification (in part) that the currently modified software is performing properly. Second, it provides a "best case" scenario (the filter model parameters exactly represent the truth model parameters) for comparison purposes. The plots of Figure A-1 show the estimation error and the LOS error for the X and Y axes with no control applied. Similarly, Figure A-2 shows the same error plots but with closed-loop control applied. Note that control is applied at the 0.5 second point. This was done to allow good state (and parameter) estimates to be achieved (after an initial transient) before they are used to generate feedback control. Clearly, the LOS deviations are quelled, not to within 1 μradian, but to a significant degree. The temporal average values of the X and Y LOS errors are 24.8 μradians and 26.1 μradians, respectively, with no control, and 4.38 μradians and 4.88 μradians, respectively, with control applied. The LOS deviations with control applied are precisely the same as that obtained by Gustafson [2: Chapter 5, pg. 5-30].

5.2.2 *Multiple Model Baseline*

The 7-point discretized parameter space developed by Gustafson [2] is used here with the true parameter set to the nominal (center) position. A bank of three filters is used, initially centered on the true parameter position. For the ME/A case, Figure A-3 shows the estimation and LOS error for the X and Y axes with control applied. The temporal average values of the X and Y LOS errors are 4.58 μradians and 5.10 μradians, respectively. For the ME/I case, Figure A-4 shows the estimation and LOS error for the X and Y axes with closed-loop control applied. The temporal average values of the X and Y LOS errors are 5.36 μradians and 6.15 μradians, respectively. It is seen here that the ME/A method performed with *15 %* and *17 %* better control on the X and Y axes, respectively. A summary of the MMAC performance is presented in Table 5-1.

Configuration	**X-axis (μradians)**	**Y-axis (μradians)**
Open-loop	24.8	26.1
Single Filter	4.38	4.88
Multiple Model (ME/A)	4.58	5.10
Multiple Model (ME/I)	5.36	6.15

Table 5-1. MMAC LOS Temporal Average RMS Deviations

5.3 *Parameter Estimation Performance*

This section presents the results from utilizing the 7-point parameter space developed by Gustafson [2]. Throughout the discussion in this section, the analysis is conducted without closed-loop control applied. The only control applied is the dither signal or the rigid-body slew command (open-loop control). In this research, the dither signals were input via the control variable, $\mathbf{u}(t_i)$ at the six PMA points on the bulkhead of the structure (locations 13 - 18). This seemed to be a logical choice for dither input, given the shape of the SPICE structure. With the larger mass located at the bulkhead, any disturbances entering the structure from the base would cause the tripod legs to be more easily shaken.

5.3.1 Baseline (No Dither)

Before evaluating the model with dither signals, part of Gustafson's work was again duplicated in order to establish a baseline to compare against. Here, two sets of simulations were run. In the first set, the true parameter is allowed to "jump" to a new value midway in the simulation. Many different configurations were tested. These included having the bank centered on the true parameter and centered at different locations in the parameter space at the start of the run. Figure A-5 is representative of these simulations with the ME/A method, which closely compares to Gustafson [2: Chapter 5, pg. 5-48]. Figure A-6 is for the ME/I case. Both of these methods provide poor parameter tracking performance, but the ME/A method displays better performance than the ME/I method. The second set of simulations consisted of allowing the true parameter to start at the 1st position and increase by one point every three seconds, up to the 7th position. Again, the bank of filters is centered at various starting positions on different runs. With the true parameter value changing in this fashion, the moving bank was unable to track the varying parameter well, under <u>any</u> bank center starting position. Figure A-7 is representative of these simulations with ME/A. Figure A-8 is the same simulation but with the ME/I method. Notice that neither method tracks the true parameter very well, but that the ME/A method does somewhat better than the ME/I. This is consistent with the better MMAC closed-loop performance with the ME/A method.

5.3.2 Dither Signal Enhancement Performance

Swept Sine Wave: Chapters 2 and 4 discussed the idea of a "best" or *identification-enhancing* dither frequency. In order to determine this frequency, a sine wave dither was applied with the dither frequency sweeping through a range of values (swept frequency sine wave). The frequencies were in the range of 7 Hz to 40 Hz. Recall from Chapter 4 that this is the range of the lower bending mode frequencies. By using this swept sine wave technique, the best dither frequency was determined to be 32 Hz, which corresponds approximately to the 11th modal frequency (see Table 4-1). Figure A-9 indicates the moving bank "zeroing in" on the true parameter value of 6. The simulation shown starts with a sine wave frequency of 29 Hz and is incremented by 1 Hz every 2 seconds. So, when a frequency of 32 Hz is reached at 6 seconds, the algorithm has "latched" on to the true parameter point value of 6. The same result of 32 Hz was obtained using both ME/A and ME/I methods. Since this identification-enhancing frequency did not correspond to precisely one of the modal frequencies, many different simulations were run in order to eliminate the possibility of there being a better dither frequency than 32 Hz. From Figure A-9, it would appear that 33 Hz would work as well, and in some situations this was true. However, a frequency of 32 Hz performed better most of the time with different signal amplitudes and with different parameter positions declared as the true value. The amplitude best suited for this model was found to be 100 Newtons (N). Small deviations from this value did not affect the results, but too small an amplitude prevented proper parameter identification, and too large a value disturbed the system to such a degree that prevented proper parameter identification as well.

Sine Wave: With the dither frequency of 32 Hz determined, the full ranging true parameter variation simulation discussed in Section 5.3.1 was accomplished with a sine wave dither input signal. Figures A-10 and A-11 show the result for the ME/A and ME/I methods, respectively. It can be seen that the ME/I method performs better than the

ME/A method. This is in contrast to the better ME/A performance with no dither applied. Note that in Figures A-10 and A-11, the parameter estimate is at times "locked" on to a value which is one off from the true parameter value. For example, in Figure A-11, for the first 3 seconds, the parameter estimate is indicating a value of 0.5, while the true parameter value is 1. The reason for this "locking" on to the incorrect parameter value may be due to the fact that the "best" dither-enhancing frequency is slightly different for each declared truth model (based on a different parameter value). Remember that the dither frequency of 32 Hz was chosen because it worked well for the entire range of parameter values.

Square Wave: The results of the square wave dither input simulations are shown in Figures A-12 and A-13 for the ME/A and ME/I methods, respectively. As with the sine wave dither signal, the ME/I method outperforms the ME/A method in parameter tracking. Many simulations were run with different square wave frequencies (similar to the swept frequency sine wave simulations). Still, the 32 Hz prevailed as the parameter-enhancing frequency. It was seen that, even with the extra high frequency content, the square wave did not adequately enhance parameter identification, when other than a 32 Hz frequency was used. Furthermore, by comparing Figure A-10 with A-12, and Figure A-11 with A-13, it was seen that there was no substantial difference between the tracking performance with sine wave or square wave dither inputs.

Wide Band Noise: Frequencies in the range of 7 Hz to 40 Hz were used for the wide band noise input signal. However, performance was degraded when frequencies above 33 Hz were used. Different values for the gain were evaluated (see Figure 2-4) with a value of 100 giving the best performance. Figure A-14 indicates the tracking performance of the ME/I method with a signal frequency range of 7 Hz to 33 Hz. This frequency range was then changed to 30 Hz to 33 Hz. Figures A-15 and A-16 show the parameter tracking performance of the ME/A and ME/I methods, respectively, for this frequency range. Again, as with the sinusoidal dither input signals, the ME/I method provided better tracking performance than the ME/A method.

5.3.3 Rigid-body Motion Enhancement Performance

Before the rigid-body matrices could be augmented to the modal matrices (see Section 3.3.3.2), a moment of inertia value had to be determined for use in Equation (3.13). From the physical description of the SPICE 2 model in the SPICE Preliminary Design Review (PDR) [14], the moment of inertia was determined to be 10000 $\mathrm{kg\,m^2}$. The simulation proceeded with a rigid-body command input of +1000 N for 1 second followed by an input of -1000 N for 1 second. These command inputs simulate the rotating structure, accelerating and then decelerating, producing a total slew angle of approximately 3.5 degrees. This is consistent with the test performed on the SPICE model as indicated in the SPICE PDR [14: IV-19] and thus helps to verify the moment of inertia calculation. Figures A-17 and A-18 show the parameter estimation performance with the slew commands just described for the ME/A and ME/I methods, respectively. As these figures indicate, the rigid-body motion causes a significant disturbance, and as the disturbance quells, the MMAE is able to track the true varying parameter value, somewhat. The large disturbance shown in these figures occurs between the 1.5 second point and the 3.5 second point (the time period the slew commands were applied). Once the rigid-body slewing motion has stopped, the algorithm takes approximately 0.5 seconds for acquisition of a good parameter estimate. Note that the excellent parameter identification performance found with purposefully constructed dither signals is not found here, but that the parameter identification is enhanced, as compared to the performance with no dither input at all. Also note that the ME/A and ME/I methods perform about the same, unlike with the purposeful excitation forms of dither enhancement.

5.4 New Parameter Space Discretization

The simulations discussed in Section 5.3 were based on the 7-point parameter space developed by Gustafson [2]. However, due to the poor performance of the parameter estimate tracking without dither enhancement, the parameter point discretization was re-investigated, as discussed in Section 4.4.3. As indicated in Chapter 4, the first parameter point obtained was only a factor of 0.0025 from the nominal point. Since it was desired to maintain the same range of undamped natural frequency factors as that used by Gustafson, this new parameter discretization would require many more than seven points. The original range of factors was 0.9811 to 1.0466. In order to conserve time, the new parameter space was determined by equally spacing each point by 0.0025. The new range used was 0.98 to 1.05. This is not the optimum parameter point spacing, but it is one with many more points than previously used, and it is easily generated. For the comparison purposes of this thesis, this will suffice. Thus, the new discretized parameter space contains 29 points (ω_1 through ω_{29}) with ω_9 as the nominal point, as indicated by Table 5-2.

Parameter	ω_1	ω_2	. . .	ω_9	. . .	ω_{29}
Factor	0.98	0.9825	. . .	1.0	. . .	1.05

Table 5-2. Parameter Space Discretization

The same set of simulations as presented in Section 5.3 were run with the new discretized parameter space to see the impact of the finer discretization. These results are presented in the next subsections.

5.4.1 Baseline (No Dither)

With these simulations, the true parameter varied from the 1st to the 29th position, changing position every second. The bank of filters is initially centered at the 9th (nominal) position. Figures A-19 and A-20 represent the baseline simulation for the ME/A and ME/I methods, respectively. Note the greatly improved tracking performance with the ME/A method versus the ME/I method. This result is, however, consistent with the result with the old parameter space tracking performance in that the ME/A method performed better with no dither enhancement.

5.4.2 Dither Signal Enhancement Performance

Sine Wave: Figures A-21 and A-22 indicate the parameter tracking performance of the ME/A and ME/I methods, respectively, with a 32 Hz, 100 N amplitude sine wave dither signal. Notice here that the performance rating for the two methods is mixed. The ME/A provides better tracking in the middle of the simulation between the 6th and 24 parameter points, while the performance of the ME/I method is better at both ends.

Square Wave: Figures A-23 and A-24 show the tracking performance of the ME/A and ME/I methods, respectively, with a square wave dither input. The mixed performance between the two methods is very similar to the performance with the sine wave dither. The results of the sine wave and square wave dither-enhanced performance are also consistent with the results from the 7-point parameter space, in that there is no substantial difference between the performance with sine or square wave dither inputs.

Wide Band Noise: Figures A-25 and A-26 indicate the parameter tracking performance of the ME/A and ME/I methods, respectively, with wide band noise frequency inputs of 30 Hz to 33 Hz. A gain value of 100 is used, as with the old parameter space simulations. Again, as with the sinusoidal dither inputs, the performance of the ME/A and ME/I methods is mixed.

Rigid-body Motion: The same rigid-body commands were simulated as before. Figures A-27 and A-28 show the parameter tracking performance of the ME/A and ME/I methods, respectively. Note that the tracking performance of both these methods is worse than with no dither input; the reason for this phenomenon is not clear.

5.5 Summary of ME/A vs ME/I Performance

Table 5-3 indicates the compilation of the comparison study accomplished with the ME/A versus the ME/I methods of probability density computation. As was shown in the

Performance Measure	Configuration	ME/A	ME/I
Multiple Model MMAC	Unchanging Parameter	√	
Parameter Tracking (7-point Space)	No Dither	√	
	Sine Wave		√
	Square Wave		√
	Wide Band Noise		√
	Rigid-body Motion	mixed	mixed
Parameter Tracking (29-point Space)	No Dither	√	
	Sine Wave		√
	Square Wave		√
	Wide Band Noise	mixed	mixed
	Rigid-body Motion	√	

Table 5-3. ME/A versus ME/I Performance (√ corresponds to better performance)

preceding discussion, neither method is better in every situation. In Table 5-3, a "√" (check) indicates which method was better in the given configuration.

5.6 Increased Measurement Precision Performance

Referring to Figure 3-7, the gain of 1.47e-4 is an indication of the variance of the truth model measurement noise. For the purpose of establishing a comparison baseline, this gain was set to zero (no measurement noise), and the single filter closed-loop control simulation of Section 5.2.1 was run. The RMS values for the X and Y LOS deviations were exactly the same as with the measurement noise included (4.38 and 4.88 μradians, respectively). Appropriate re-tuning of the Kalman filter **R** matrix values was attempted, with no change in the result. In fact, the **R** matrix values could not be reduced by any significant amount due to the $\left[\mathbf{H}(t_i)\mathbf{P}(t_i^-)\mathbf{H}^T(t_i)+\mathbf{R}(t_i)\right]$ term in Equation (2.19) becoming "too small", preventing the steady state Kalman filter gain from being computable due to difficulty in inverting the $\left[\mathbf{H}(t_i)\mathbf{P}(t_i^-)\mathbf{H}^T(t_i)+\mathbf{R}(t_i)\right]$ matrix. Thus, increased measurement precision had no effect on the MMAC performance. Clearly, more information in the form of additional measurements, and/or additional actuators, must be included before desired controller performance will be realized.

5.7 Reduced-Order Design Performance

The reduced-order design model was constructed according to the modal reduction technique described in Chapter 2. The modal reduced design consisted of the first 26 bending modes (52 bending mode states) plus 54 PMA and accelerometer states, for a total of 106 states. This 106-state reduced-order filter was run against the 194-state truth model in the single filter configuration with closed-loop control applied. Figure A-29 shows the estimation and LOS error plots with closed-loop control applied. Despite the relatively high number of bending mode states included in this filter design, the algorithm was unable to provide adequate control to quell the vibrating structure. The temporal averages for the X and Y LOS errors were 21.0 and 23.3 μradians, respectively. The result is the same as that obtained by Gustafson [2: Chapter 5, pg. 5-21] with his 6 bending mode (66-state) reduced-order filter model.

5.8 Summary

This chapter has presented the results from the analysis conducted in this research. First presented was the artificially informed single filter performance baseline, along with a verification of Gustafson's [2] single-filter result. For comparison purposes, the multiple model adaptive algorithm performance with an unchanging parameter value was also shown. The results of implementing the various dithering techniques were then presented. The results indicated the excellent parameter enhancement with the sine and square wave dither signals. The wide band noise dither signal was also found to provide effective enhanced parameter identification. The results of naturally exciting the structure through the use of rigid-body slew commands did improve the performance of parameter identification, although not to the degree exhibited with the purposeful dither excitation. This was followed by a presentation of results of parameter identification utilizing the new discretized parameter space. The striking difference in performance with the new, more finely discretized parameter space was the very good parameter tracking, with no dither enhancement. Improved parameter identification was also obtained with dither enhancement. Throughout the discussion of MMAE/MMAC and parameter estimation tracking performance, a comparison was made of the ME/A and ME/I methods of hypothesis conditional probability computation. As was shown in Table 5-3, neither method is better than the other in every situation. Lastly, the results of increased measurement precision and reduced-order filter design were shown. Although these last two areas of investigating did not show the "positive" results as that obtained with the other areas of this thesis research, the results obtained did shed light into the limitations of the SPICE 2 model. Chapter 6 will discuss the conclusions drawn from analyzing the results displayed in this chapter, as well put forth some recommendations for further research in this area.

6. Conclusions and Recommendations

6.1 Introduction

The main purpose of this thesis has been to research and design effective dither inputs for enhanced parameter identifiability for the SPICE 2 flexible space structure. Previous work by Gustafson [2] indicated the need for enhanced parameter identification. Thus, this research with the SPICE 2 structure, utilizing moving-bank Multiple Model Adaptive Estimation and Control (MMAE/MMAC) algorithms, has been a direct continuation of Gustafson's work. Purposeful excitation dither inputs consisting of sinusoidal waves, square waves, and wide band noise were investigated, as well as purposeful rigid-body slew commands. The rigid-body motion was applied to determine the effects on enhanced parameter identification through natural excitation of the structure's bending modes. Other issues explored in this research were: (1) a new discretized parameter space, (2) increased measurement precision, and (3) a reduced-order filter/controller.

The ultimate conclusions reached in the course of this research are: (1) dither-enhanced parameter estimation performs very well for the SPICE 2 flexible structure, and (2) better measurement precision and reduced-order filter/controller designs (up to 26 structure modes) do not yield adequate state estimation and control performance for the MMAE/MMAC algorithms.

6.2 Conclusions

The analysis conducted in this research began with a verification of Gustafson's work with the single filter/controller design. This was followed by establishing baseline performances with the 7-point parameter space, consisting of performance against (1) an unchanging parameter value and (2) a varying parameter value, both with no dither enhancement. Here, it was seen that the ME/A method performed better than the ME/I method of hypothesis conditional probability computation. In contrast, all of the performance evaluations conducted with purposefully selected dither inputs indicated that the ME/I method provided better parameter identification than the ME/A method. Of the various forms of dither inputs evaluated for parameter identification performance, each performed very well. While the sinusoid dither signals were better able to "latch" on to the true parameter value, the best parameter-identification dither frequency had to be correctly determined. With the wide band noise dither input, the "latching" on effect was not seen, however, a broad range of dither frequencies could be used with very good performance.

Thus, each type of dither input has its advantage, and a combination of the two types could be used in a real application. For example, wide band noise dither could help identify the correct range of frequencies. The noise band could be varied to the point that the parameter-identification frequency would be known. Then, a switch to a sinusoid dither at this frequency could be made.

Parameter identifiability enhancement was also observed from the effects of rigid-body rotation-induced excitation of the bending modes. However, the effects from this type of commanded slew input were not as pronounced as those from the sinusoid and wide band noise dither inputs. Indeed, parameter tracking was improved with rigid-body slewing and could possibly be exploited in a real situation whenever the structure is moved (pointed) in a different direction.

The conclusion drawn from simulations conducted with the new 29-point discretized parameter space is that this new parameter space is a viable alternative to the 7-point space, if no dither enhancement is to be used. With dither-enhanced parameter identification, the two parameter space discretizations seemed to perform equally well. Thus, dither enhancement could be used with the 7-point parameter space as an alternative to the larger memory storage requirement with the 29-point parameter space.

The analyses performed with the increased measurement precision and the reduced-order filter/controller revealed some limitations with the SPICE 2 model. The conclusions drawn from these results are: (1) more and/or different types of measurements may be necessary for improved MMAC performance, (2) better and/or additional actuators should be considered.

6.3 Recommendations

Dither-enhanced parameter identification was shown by this research to be very effective with the moving-bank MMAE/MMAC algorithms, as applied to the SPICE 2 flexible structure. Recommendations for further research are:

1. Implement the latest SPICE structure model.
2. Apply the dither-enhancing methods to other multiple model adaptive research.
3. Investigate the poor parameter tracking performance of the 29-point parameter space with rigid-body rotation-induced excitation of the bending modes.
4. Consider more than just a single scalar multiplier on bending mode frequencies as the uncertain parameter. Use a vector of parameters, thereby allowing for different *relative* spacing of the modal frequencies as well as their simultaneous movement on the frequency axis.

Appendix A: MMAE, MMAC, and Parameter Estimation Performance Plots

This appendix contains all the plots corresponding to the results discussion in Chapter 5. Figures A-1 through A-4 and A-29 contain mean ± one standard deviation ($\pm\sigma$) plots of estimation error and LOS error. Each of these four figures contain four plots, labeled (a), (b), (c), and (d). (a) and (b) correspond to the X- and Y-axis estimation error plots; (c) and (d) correspond to the X- and Y-axis LOS error plots. The remaining figures are plots of the mean parameter estimate. Shown also on these plots is the true parameter value, for comparison.

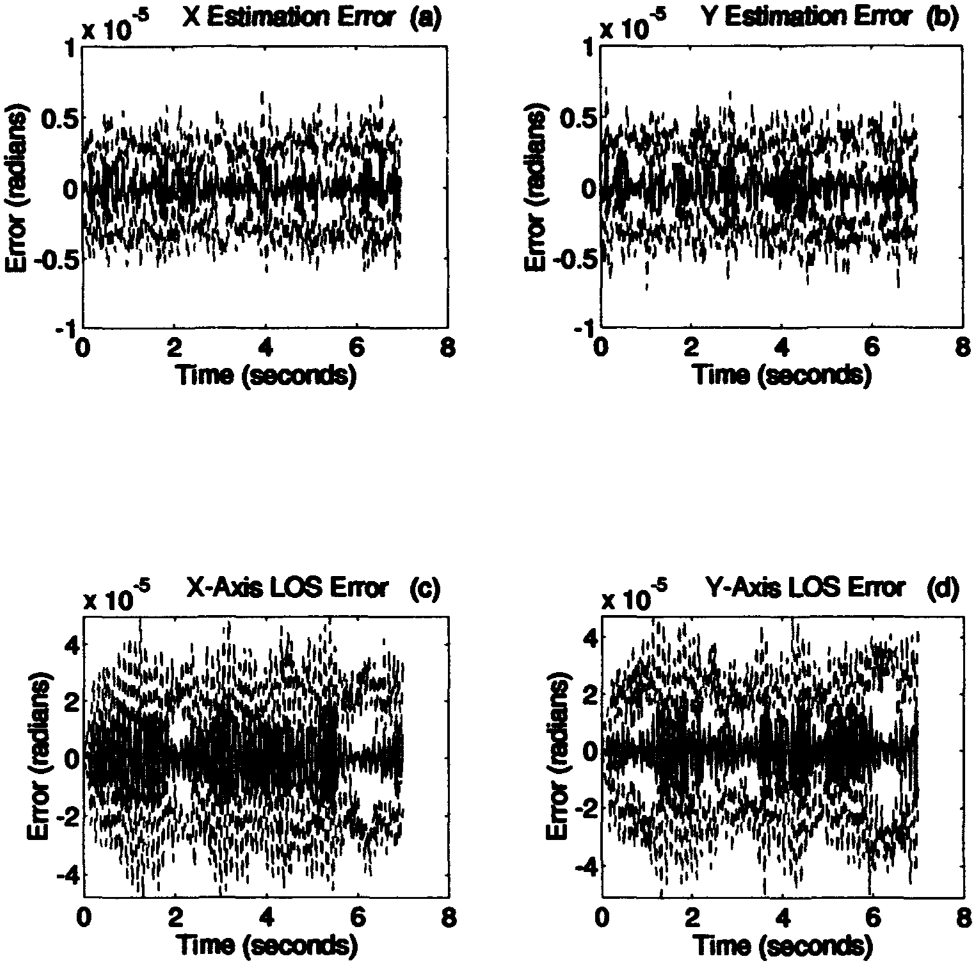

Figure A-1. Single Filter Estimation and Control Baseline (Open Loop)

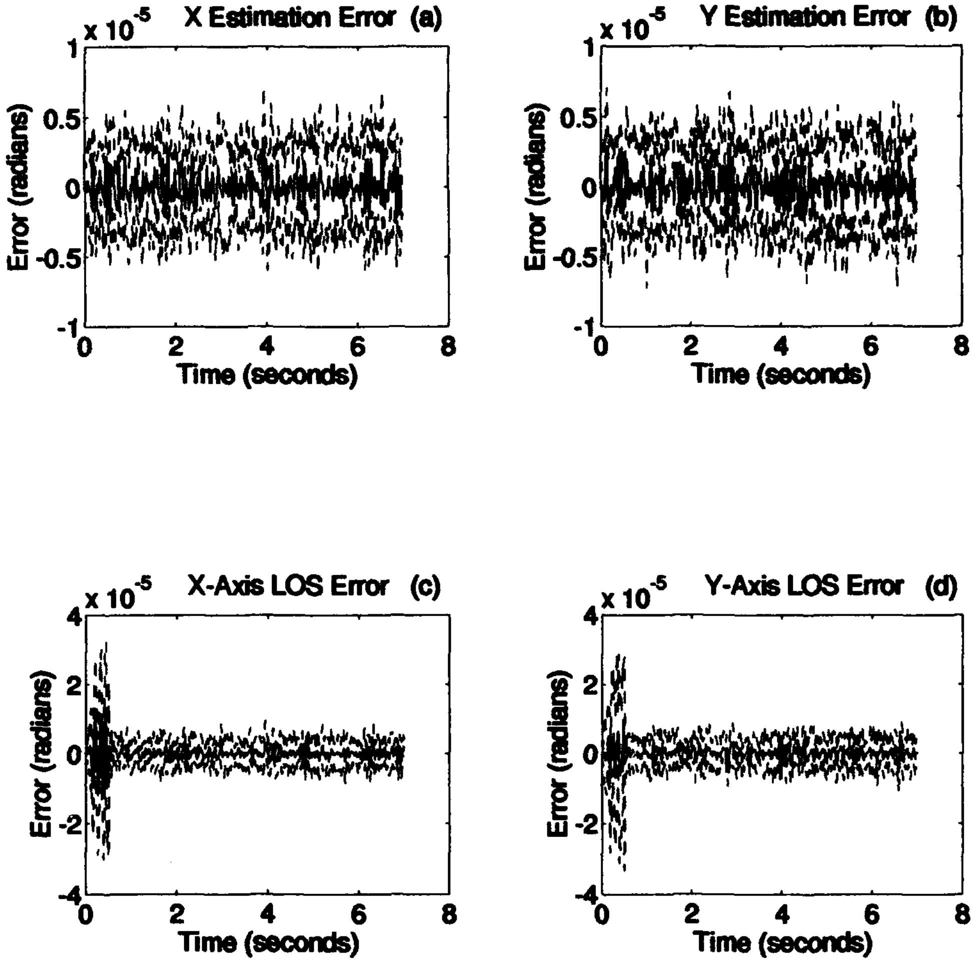

Figure A-2. Single Filter Estimation and Control Baseline (Closed Loop)

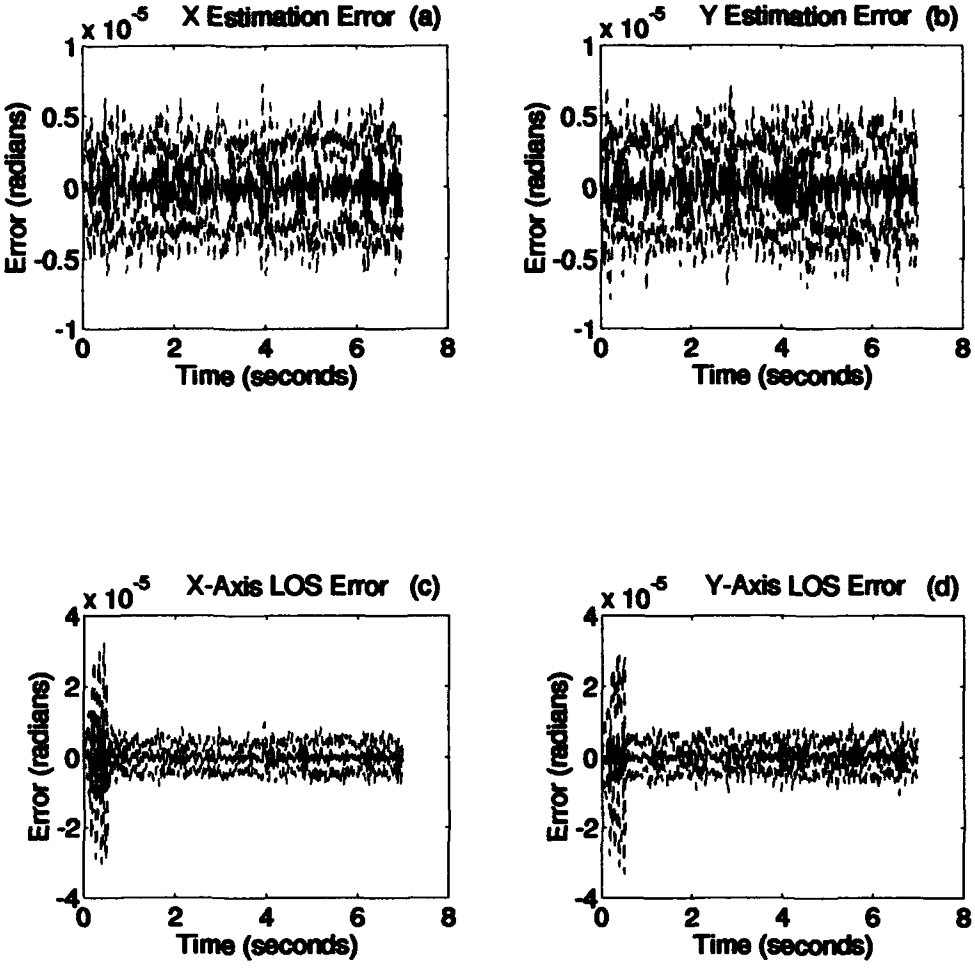

Figure A-3. Multiple Filter Estimation and Control Baseline (Closed Loop, ME/A)

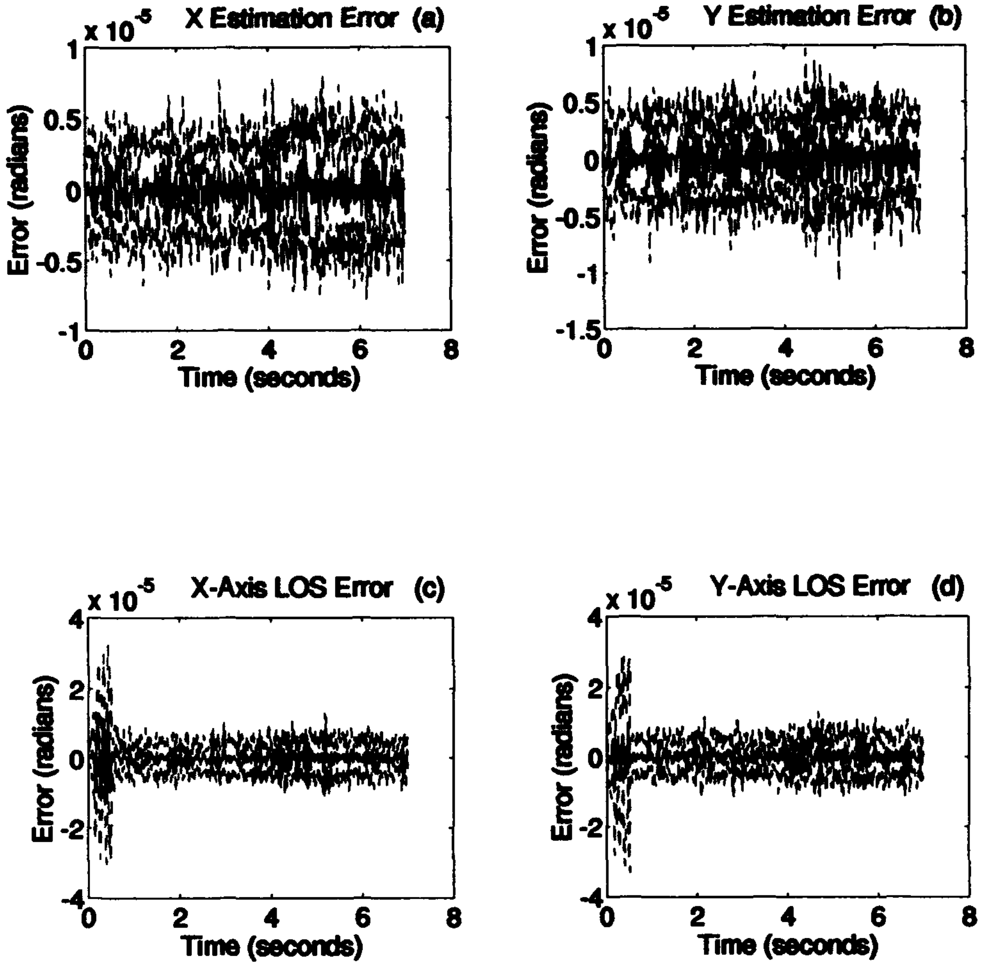

Figure A-4. Multiple Filter Estimation and Control Baseline (Closed Loop, ME/I)

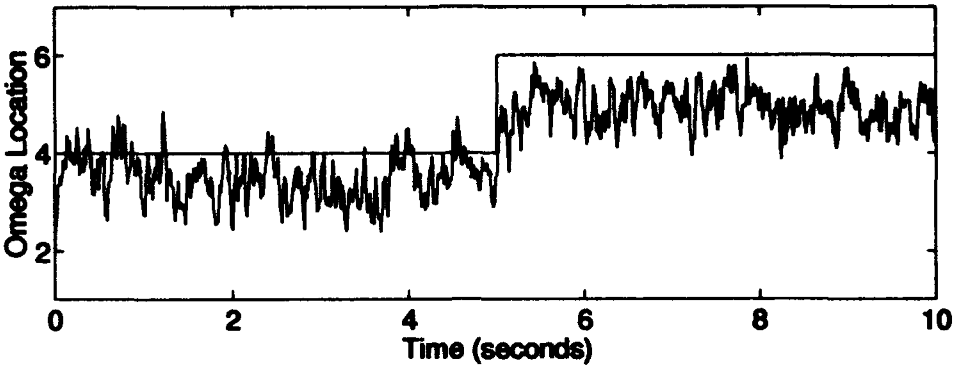

Figure A-5. 7-Point Parameter Space: True Parameter Jump, No Dither (ME/A)

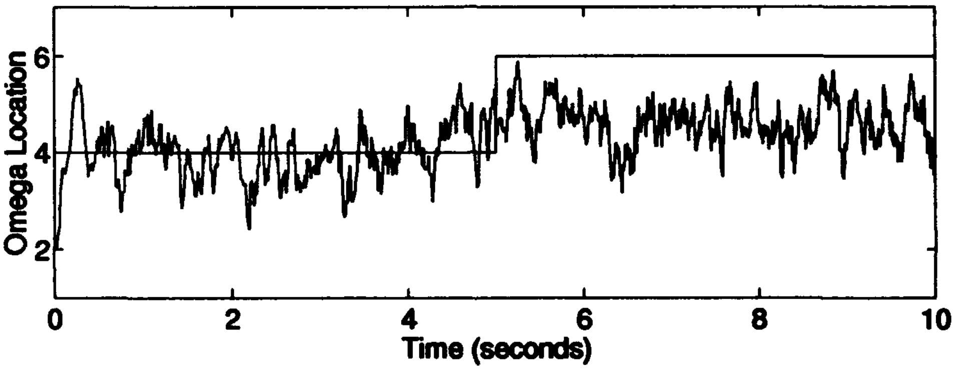

Figure A-6. 7-Point Parameter Space: True Parameter Jump, No Dither (ME/I)

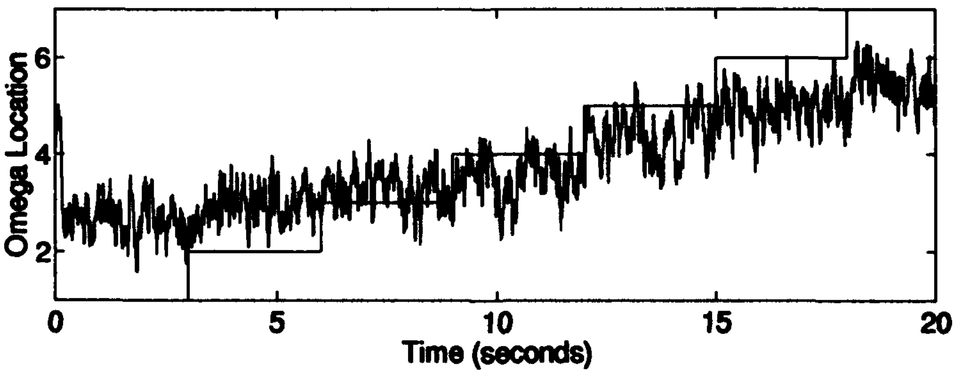

Figure A-7. 7-Point Parameter Space: Parameter Variation, No Dither (ME/A)

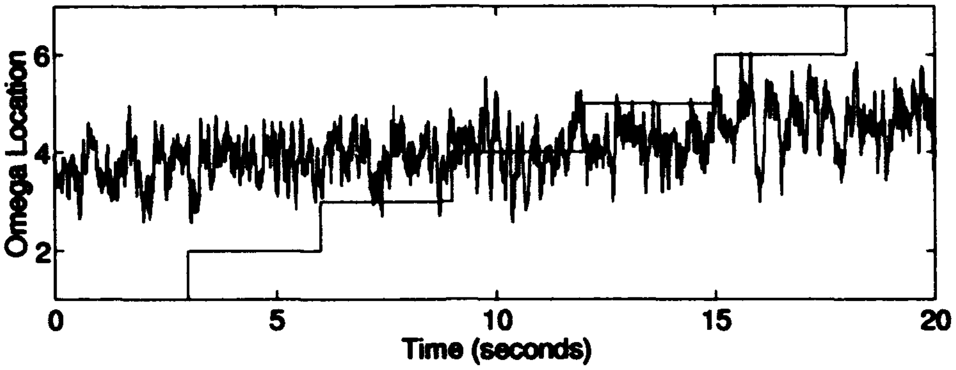

Figure A-8. 7-Point Parameter Space: Parameter Variation, No Dither (ME/I)

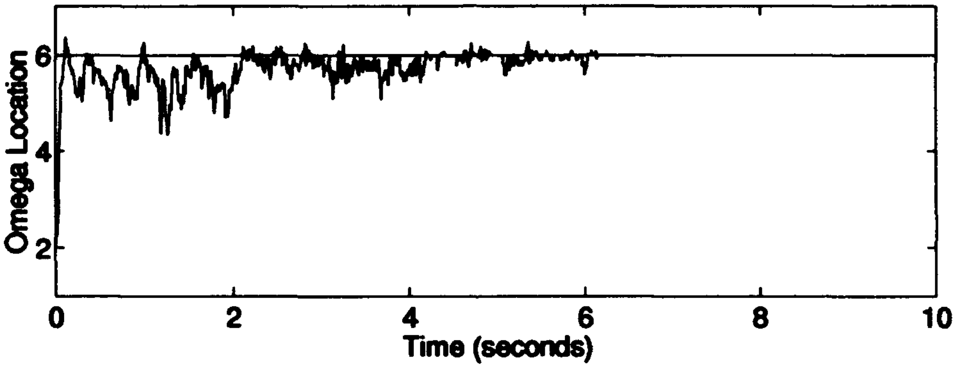

Figure A-9. 7-Point Parameter Space: Swept Sine Wave Dither, 29-33 Hz (ME/A)

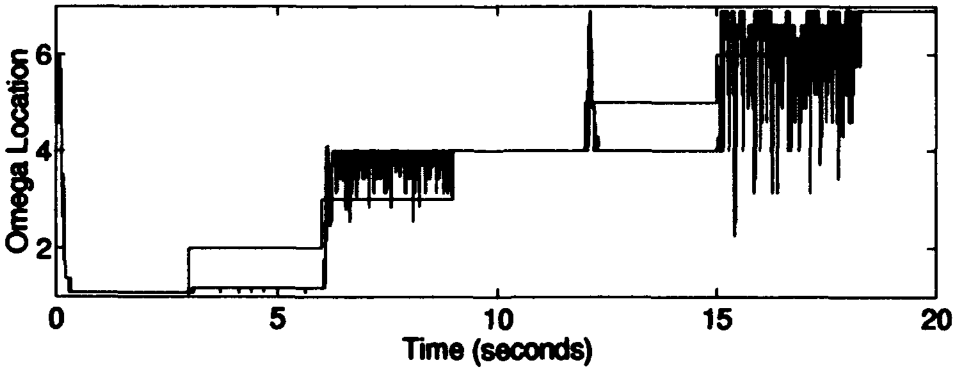

Figure A-10. 7-Point Parameter Space: Sine Wave Dither (ME/A)

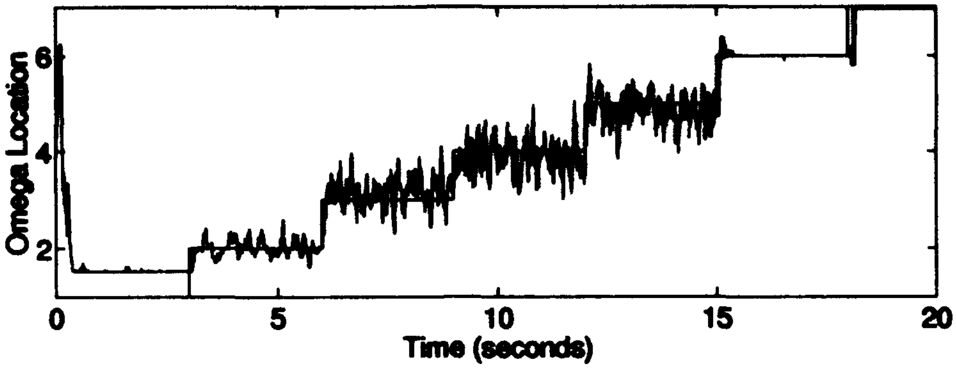

Figure A-11. 7-Point Parameter Space: Sine Wave Dither (ME/I)

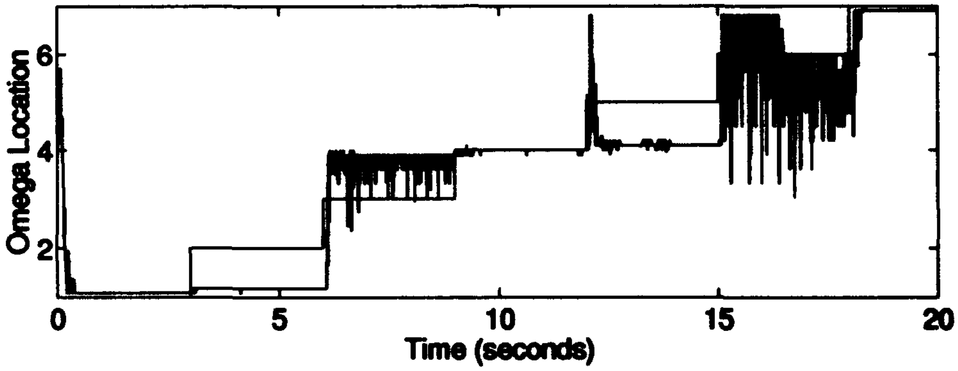

Figure A-12. 7-Point Parameter Space: Square Wave Dither (ME/A)

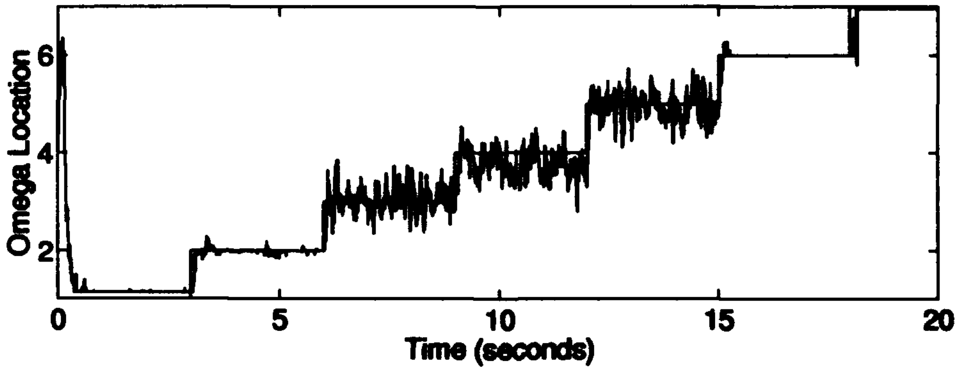

Figure A-13. 7-Point Parameter Space: Square Wave Dither (ME/I)

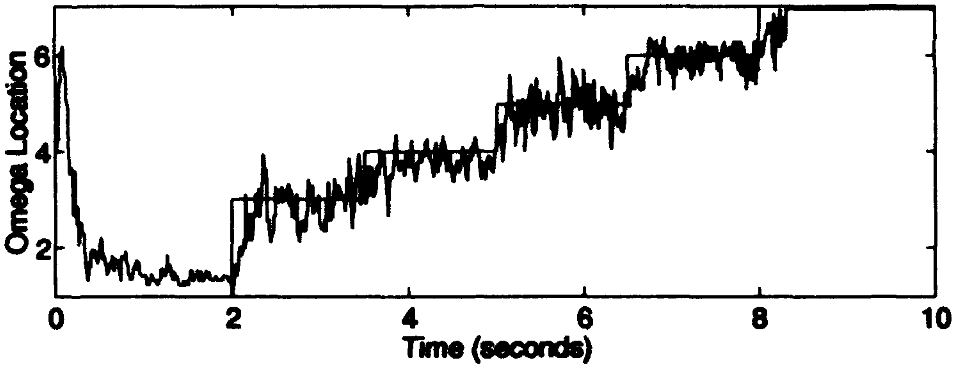

Figure A-14. 7-Point Parameter Space: Wide Band Noise Dither, 7-33 Hz (ME/I)

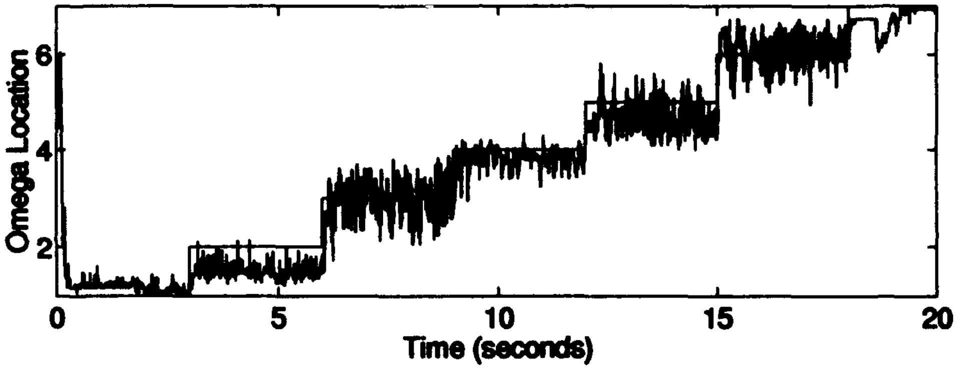

Figure A-15. 7-Point Parameter Space: Wide Band Noise Dither, 30-33 Hz (ME/A)

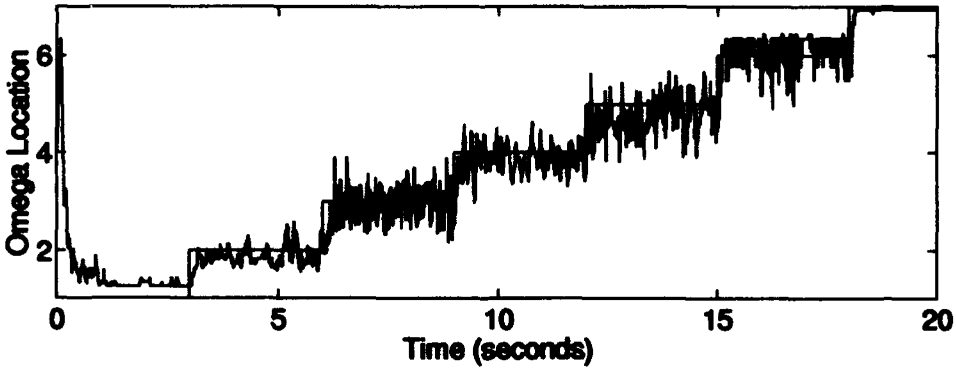

Figure A-16. 7-Point Parameter Space: Wide Band Noise Dither, 30-33 Hz (ME/I)

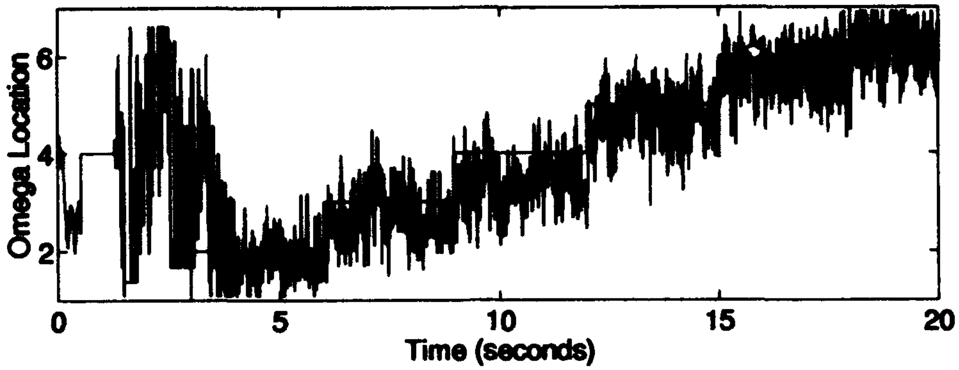

Figure A-17. 7-Point Parameter Space: Rigid-body Dither (ME/A)

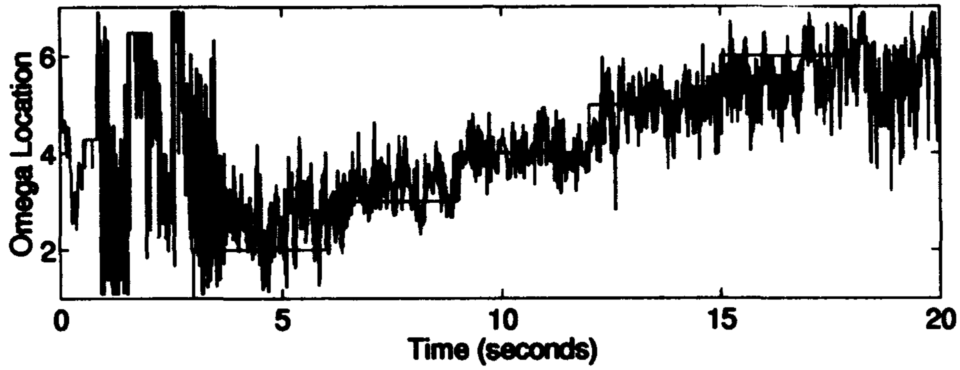

Figure A-18. 7-Point Parameter Space: Rigid-body Dither (ME/I)

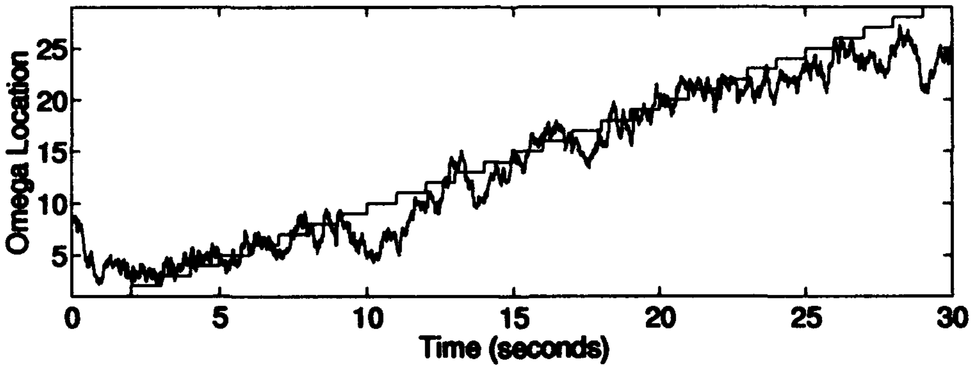

Figure A-19. 29-Point Parameter Space: No Dither (ME/A)

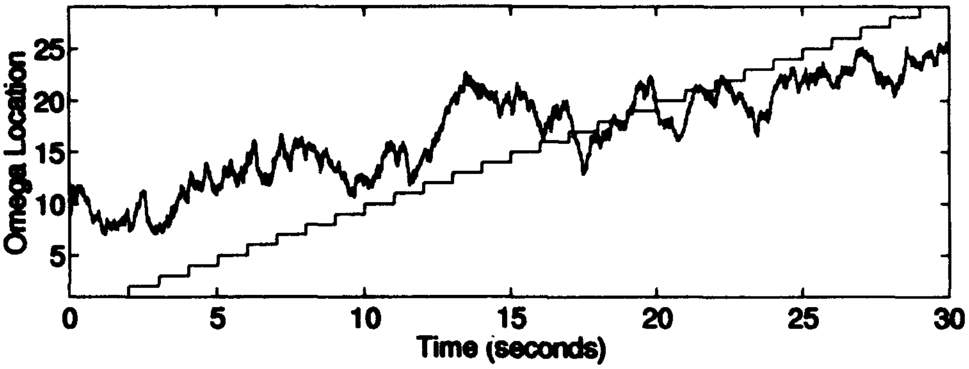

Figure A-20. 29-Point Parameter Space: No Dither (ME/I)

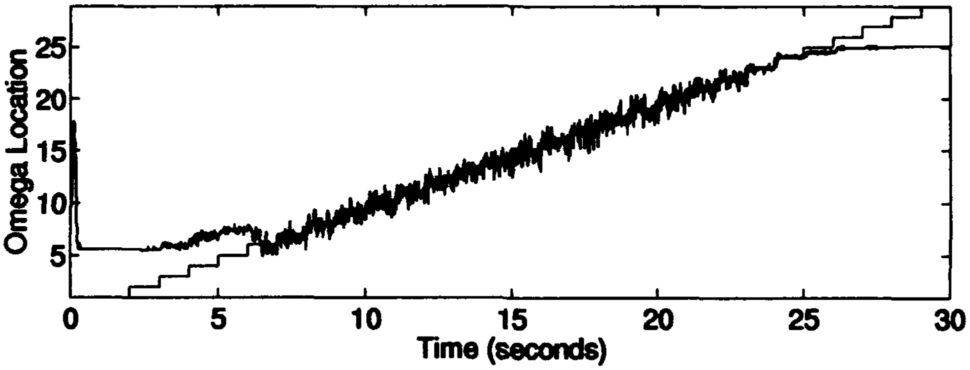

Figure A-21. 29-Point Parameter Space: Sine Wave Dither (ME/A)

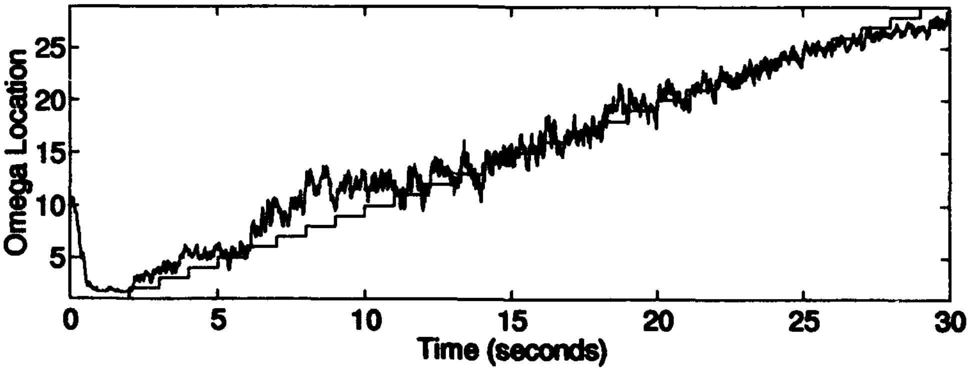

Figure A-22. 29-Point Parameter Space: Sine Wave Dither (ME/I)

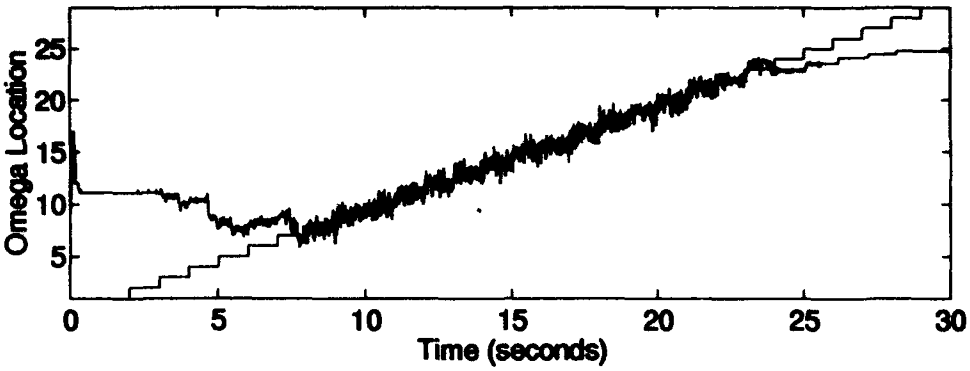

Figure A-23. 29-Point Parameter Space: Square Wave Dither (ME/A)

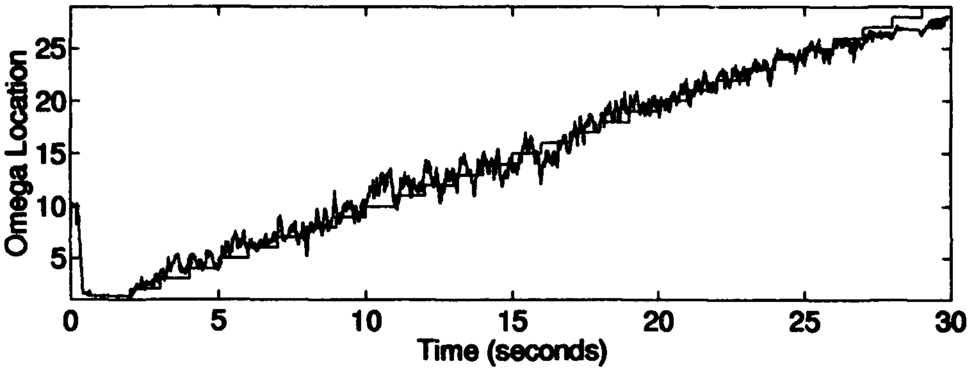

Figure A-24. 29-Point Parameter Space: Square Wave Dither (ME/I)

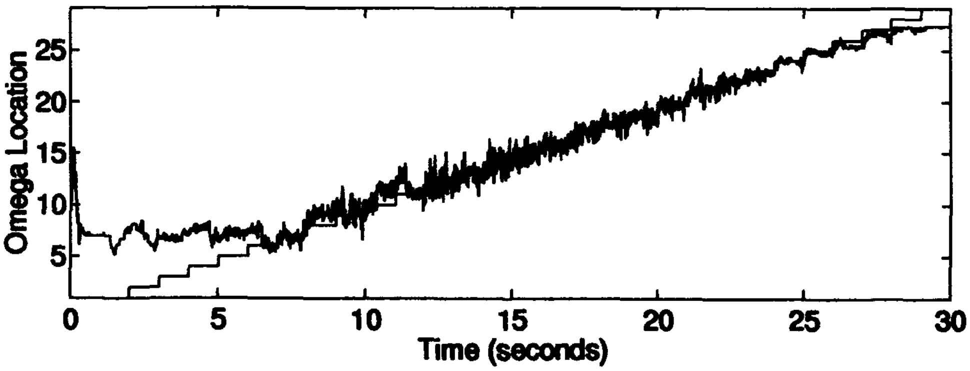

Figure A-25. 29-Point Parameter Space: Wide Band Noise Dither, 30-33 Hz (ME/A)

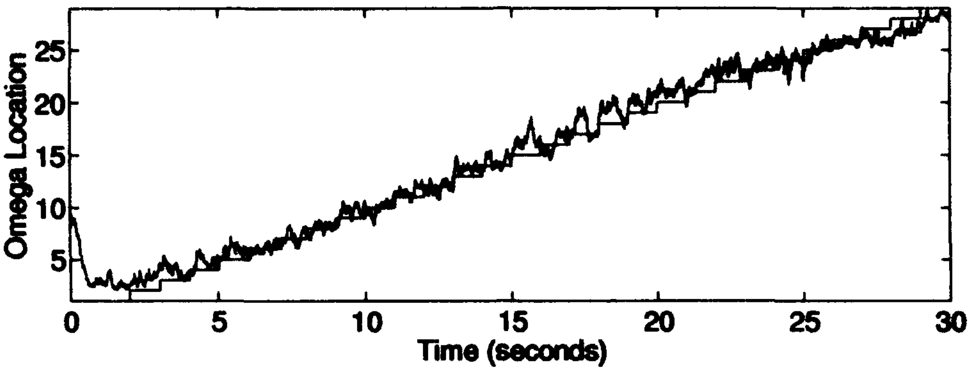

Figure A-26. 29-Point Parameter Space: Wide Band Noise Dither, 30-33 Hz (ME/I)

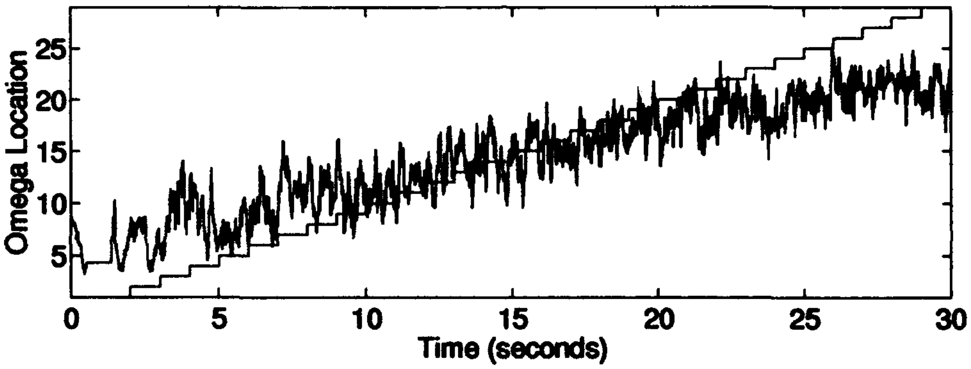

Figure A-27. 29-Point Parameter Space: Rigid-body Dither (ME/A)

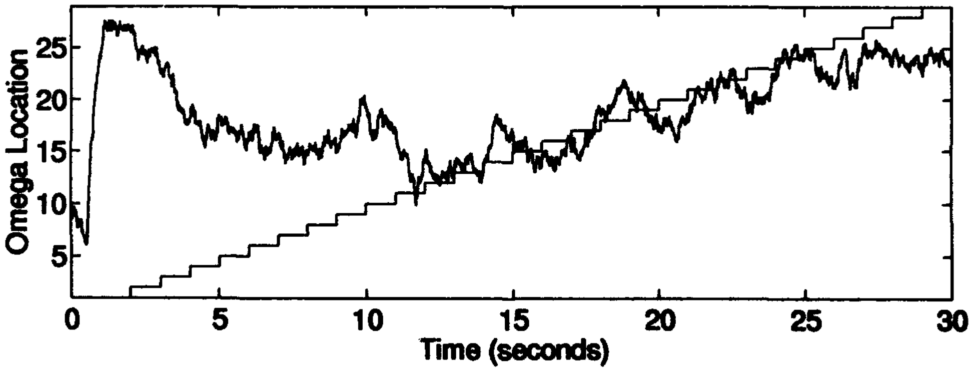

Figure A-28. 29-Point Parameter Space: Rigid-body Dither (ME/I)

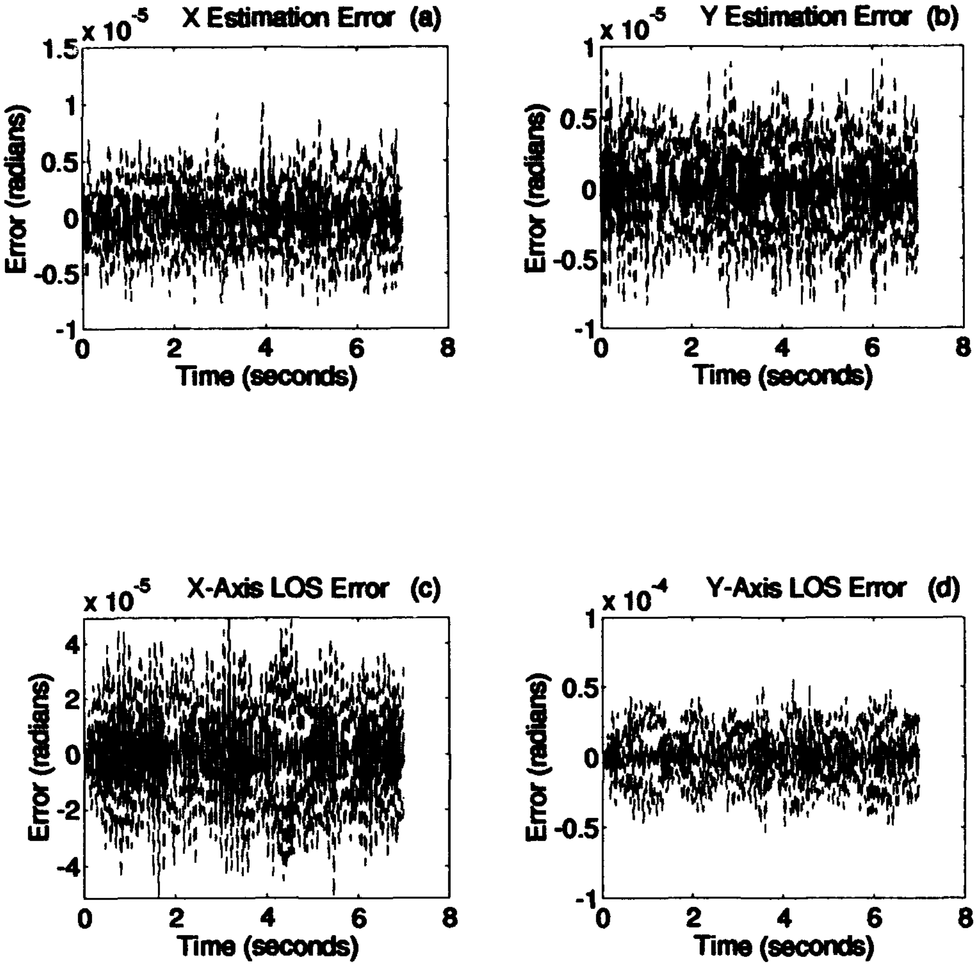

Figure A-29. Single Filter Estimation and Control, 26-Mode Reduced Order (Closed Loop)

Bibliography

1. Filios, Capt Paul G. *Moving-Bank Multiple Model Adaptive Algorithms Applied to Spacecraft Control.* MS Thesis AFIT/GE/ENG/85D-14. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB, OH, December 1985 (AD-A164 016).

2. Gustafson, Capt John A. *Control of a Large Flexible Space Structure Using Multiple Model Adaptive Algorithms.* MS Thesis AFIT/GE/ENG/91D-22. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB, OH, December 1991. (AD-A243 759).

3. Gustafson, Capt John A. and Peter S. Maybeck. "Control of a Large Flexible Space Structure with Moving-Bank Multiple Model Adaptive Algorithms," *Proceedings of the 31st IEEE Conference on Decision and Control*: 1273-1278. Tucson, AZ (December 1992).

4. Hanlon, Capt Peter D. *Failure Identification Using Multiple Model Adaptive Estimation for the Lamda Flight Vehicle.* MS Thesis AFIT/GE/ENG/92D-19. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB, OH, December 1992.

5. Hentz, 1Lt Karl P. *Feasibility Analysis of Moving Bank Multiple Model Adaptive Estimation and Control Algorithms.* MS Thesis AFIT/GE/ENG/84D-32. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB, OH, December 1984 (AD-A152 015).

6. IMSL Math/Library: FORTRAN Subroutines for Mathematical Applications. IMSL, Inc., Houston, Texas, January 1989.

7. Karnick, 2Lt Drew A. *Moving Bank Multiple Model Adaptive Estimation Applied to Flexible Spacestructure Control.* MS Thesis AFIT/GE/ENG/86D-41. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB, OH, December 1986 (AD-A178 870).

8. Karnick, Drew A., and Peter S. Maybeck. "Moving Bank Multiple Model Adaptive Estimation Applied to Flexible Spacestructure Control," *Proceedings of the 26th IEEE Conference on Decision and Control*: 1249-1257. Los Angeles, CA (December 1987).

9. Kokotovic, P. V., R. E. O'Malley, Jr., and P. Sannuti. "Singular Perturbations and Order Reduction in Control Theory--An Overview," *Automatica, 12*: 123-132, 1976.

10. Kreutz, Kenneth K. *Stability and Convergence of Parallel Adaptive Regulators: Towards a Robust Perspective.* PhD Thesis, University of California, San Diego, CA, 1985.

11. Lashlee, Capt Robert W. *Moving Bank Multiple Model Adaptive Estimation Applied to Flexible Spacestructure Control.* MS Thesis AFIT/GE/ENG/87D-36. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB, OH, December 1987.

12. Lashlee, Robert W., and Peter S. Maybeck. "Spacestructure Control Using Moving Bank Multiple Model Adaptive Estimation," *Proceedings of the 27th IEEE Conference on Decision and Control*: 712-717. Austin, TX (December 1988).

13. Lynch, P. J., and Siva S. Banda. "Active Control for Vibration Damping," *Damping: 1986 Proceedings*, Technical Report. Flight Dynamics Laboratory, Air Force Wright Aeronautical Laboratory, Wright-Patterson AFB, OH, 1986 (AFWAL-TR-86-3509).

14. Lockheed et al. "Preliminary Design Review: Space Integrated Controls Experiment." Subtask 02-03. 21-22 February 1991.

15. MATLAB: Pro-MATLAB for Sun Workstations. The MathWorks, Inc., South Natick, MA, January 1990.

16. MATRIXx: ANALYSIS AND DESIGN MANUAL. "A CAD/CAE Program," Integrated Systems Inc., Santa Clara, CA, April 1990.

17. Maybeck, Peter S. Class handout distributed in MMAE special study session. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB, OH, December 1992.

18. Maybeck, Peter S. Professor. Personal interviews. School of Engineering. Air Force Institute of Technology (AU). Wright-Patterson AFB, OH, March through November 1993.

19. Maybeck, Peter S. *Stochastic Models, Estimation, and Control, Volume 1.* NY: Academic Press, 1979.

20. Maybeck, Peter S. *Stochastic Models, Estimation, and Control, Volume 2.* NY: Academic Press, 1982.

21. Maybeck, Peter S. *Stochastic Models, Estimation, and Control, Volume 3.* NY: Academic Press, 1982.

22. Maybeck, Peter S., and Capt Karl P. Hentz. "Investigation of Moving-Bank Multiple Model Adaptive Algorithms," *AIAA Journal of Guidance, Control and Dynamics* 10:1 90-96 (Jan-Feb 1987).

23. Maybeck, Peter S., and Michael Roger Schore. "Reduced-Order Multiple Model Adaptive Controller for Flexible Spacestructure," *IEEE Transactions on Aerospace and Electronic Systems*, AES-28(3): 756-767 (July 1992).

24. Menke, Timothy E. *Multiple Model Adaptive Estimation Applied to the Vista F-16 with Actuator and Sensor Failures.* MS Thesis AFIT/GA/ENG/92J-01. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB, OH, June 1992.

25. Menke, Timothy E. and Peter S. Maybeck. "Sensor/Actuator Failure Detection in the VISTA F-16 by Multiple Model Adaptive Estimation," *Proceedings of the 1993 American Control Conference*: 3135-3140. San Francisco, CA (June 1993).

26. Moyle, Capt Robert Brent. *Moving-Bank Multiple Model Adaptive Estimation and Control to a Large Space Structure*. MS Thesis AFIT/GE/ENG/90D-45. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB, OH, December 1990.

27. Roden, Martin S. *Digital and Data Communication Systems*. Englewood Cliffs, NJ: Prentice-Hall, Inc. 1982.

28. Schiller, Gregory S. *Control of a Large Flexible Space Structure Using Multiple Model Adaptive Control Techniques*. MS Thesis AFIT/GA/ENG/93D-02. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB, OH, December 1993.

29. Schore, Capt Michael Roger. *Robustness of a Moving-Bank Multiple Model Adaptive Controller for a Large Space Structure*. MS Thesis AFIT/GE/ENG/89D-46. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB, OH, December 1989.

30. Sheldon, Stuart N. *An Optimizing Design Strategy for Multiple Model Adaptive Estimation and Control*. PhD dissertation. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB, OH, December 1989.

31. Sheldon, Stuart N., and Peter S. Maybeck. "An Optimizing Design Strategy for Multiple Model Adaptive Estimation and Control," *IEEE Transactions on Automatic Control*, AC-38(4): 651-654 (April 1993).

32. Van Der Werken, Capt Daniel F., Jr. *A Robustness Analysis of Moving-Bank Multiple Model Adaptive Estimation and Control of a Large Flexible Space Structure*. MS Thesis AFIT/GE/ENG/88D-59. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB, OH, December 1988.

33. Venkayya, V. B. and V. A. Tischler. "Frequency Control and Its Effects on the Dynamic Response of Flexible Structures," *AIAA Journal 23*(11): 1768-1774 (November 1985).

Vita

Captain James A. Fitch was born on 11 October 1962 in Raymondville, Texas. He graduated from Raymondville High School in 1981. In the summer of 1981 he entered the Air Force and attended Basic Military Training School (BMTS), Lackland AFB, TX. From there he attended the Electronics Technical School at Keesler AFB, MS. His first assignment was at Langley AFB, VA where he worked as an electronics technician for the 1913th Information Systems Group. After being selected for the Airman's Education and Commissioning Program (AECP), he completed his Bachelor of Science degree in Astronautical Engineering at Mississippi State University, MS in May 1988. In September 1988 he graduated from Officer Training School (OTS), Lackland AFB, TX with a reserve commission. His next assignment was with the 5th Satellite Control Squadron, Fairchild AFB, WA as a Satellite Systems Engineer for the Defense Meteorological Satellite Program (DMSP). In May 1991 he entered the School of Engineering, Air Force Institute of Technology.

Permanent address: 345 E. Louisiana Ave.
Raymondville, TX 78580

REPORT DOCUMENTATION PAGE	Form Approved OMB No. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE December 1993	3. REPORT TYPE AND DATES COVERED Master's Thesis

4. TITLE AND SUBTITLE Multiple Model Adaptive Control of A Large Flexible Space Structure with Purposeful Dither for Enhanced Identifiability	5. FUNDING NUMBERS
6. AUTHOR(S) James Alan Fitch, Captain, USAF	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Air Force Institute of Technology, Wright-Patterson AFB, OH, 45433-6583	8. PERFORMING ORGANIZATION REPORT NUMBER AFIT/GA/ENG/93D-01
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) Phillips Laboratory/VTSA, Kirtland AFB, NM 87117-5776	10. SPONSORING/MONITORING AGENCY REPORT NUMBER
11. SUPPLEMENTARY NOTES	
12a. DISTRIBUTION/AVAILABILITY STATEMENT Approved for public release; distribution unlimited	12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

Dithering techniques for enhancing uncertain parameter identification with moving-bank MultipleModel Adaptive Estimation (MMAE) and Control (MMAC) algorithms are analyzed in this thesis. The dithering techniques and multiple-model adaptive algorithms are applied to the SPICE 2 flexible space structure.

The dithering techniques studied include purposefully constructed square wave, sine wave (fixed and swept frequency) and wide-band noise wave forms. Purposeful rigid-body slew commands are also used in order to excite the structure's flexible bending modes. Dither inputs into the structure are performed in an effort to enhance the open-loop identifiability of the uncertain parameter, namely a scalar multiplier on the undamped natural frequencies of the flexible bending modes. Correct identification of this parameter enables the MMAE and MMAC algorithms to quell any vibrations induced into the structure, in the face of varying parameter values.

The results of this study indicate that purposefully constructed dither signals do enhance parameter identification significantly. The parameter identification enhancement due to rigid-body slew command input was shown to be less effective than that with sinusoid and wide band noise dither inputs, however.

14. SUBJECT TERMS Multiple Model Adaptive Estimation, Multiple Model Adaptive Control, LQG Control, Flexible Structure, Parameter Identification, Dither Techniques	15. NUMBER OF PAGES 141
	16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT UL